|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikator** | **Raven** | **Razred** | **Standard** | **Pojasnilo** | **Pojem** | **Podkoncept** | **Praksa** | **Podpraksa** |
| 1A-CS-01 | 1A | K-2 | **Izberite in upravljajte ustrezno programsko opremo za izvajanje različnih opravil in se zavedajte, da imajo uporabniki različne potrebe in želje glede tehnologije, ki jo uporabljajo.** | Ljudje uporabljajo računalniške naprave za natančno in hitro opravljanje različnih nalog. Študenti bi morali imeti možnost izbrati ustrezno aplikacijo / program, ki ga bodo uporabili za naloge, ki jih morajo opraviti. Na primer, če se od učencev zahteva, da narišejo sliko, bi morali imeti možnost, da odprejo in uporabijo aplikacijo / program za risanje za dokončanje te naloge, ali če jih prosimo, da ustvarijo predstavitev, bi morali biti sposobni odpreti in uporabljati predstavitveno programsko opremo. Poleg tega bi morali učenci z vodstvom učiteljev primerjati in razpravljati o preferencah za programsko opremo z enako primarno funkcionalnostjo. Študenti so lahko primerjali različne spletne brskalnike ali programe za obdelavo besedil, predstavitev ali risanje. | Računalniški sistemi | Naprave | Spodbujanje vključujoče računalniške kulture | 1.1 |
| 1A-CS-02 | 1A | K-2 | **Uporabite ustrezno terminologijo pri prepoznavanju in opisovanju funkcije skupnih fizičnih komponent računalniških sistemov (strojne opreme).** | Računalniški sistem je sestavljen iz strojne in programske opreme. Strojna oprema je sestavljena iz fizičnih komponent. Študenti bi morali biti sposobni prepoznati in opisati funkcijo zunanje strojne opreme, kot so namizni računalniki, prenosni računalniki, tablični računalniki, monitorji, tipkovnice, miške in tiskalniki. | Računalniški sistemi | Strojna in programska oprema | Komuniciranje o računalništvu | 7.2 |
| 1A-CS-03 | 1A | K-2 | **Opišite osnovne težave s strojno in programsko opremo z uporabo natančne terminologije.** | Težave z računalniškimi sistemi imajo različne vzroke. Študentom na tej ravni ni treba razumeti teh vzrokov, vendar bi morali biti sposobni sporočiti težavo s točno terminologijo (npr. Ko aplikacija ali program ne deluje po pričakovanjih, naprava se ne vklopi, zvok ne deluje itd.). V idealnem primeru bi študenti lahko uporabili preproste strategije za odpravljanje težav, vključno z izklopom in vklopom naprave za ponovni zagon, zapiranjem in ponovnim odpiranjem aplikacije, vklopom zvočnikov ali priključitvijo slušalk. Te pa niso določene v standardu, ker se te težave morda ne bodo pojavile. | Računalniški sistemi | Odpravljanje težav | Preizkušanje in izpopolnjevanje računalniških artefaktov, komuniciranje o računalništvu | 6.2, 7.2 |
| 1A-NI-04 | 1A | K-2 | **Pojasnite, kaj so gesla in zakaj jih uporabljamo, ter uporabite močna gesla za zaščito naprav in podatkov pred nepooblaščenim dostopom.** | Učenje zaščite naprave ali informacij pred neželeno uporabo s strani drugih je bistven prvi korak pri učenju o kibernetski varnosti. Študentom ni treba uporabljati več močnih gesel. Ustrezno bi morali uporabljati in zaščititi gesla, ki jih morajo uporabljati. | Omrežja in internet | Kibernetska varnost | Komuniciranje o računalništvu | 7.3 |
| 1A-DA-05 | 1A | K-2 | **Shranjujte, kopirajte, iskajte, pridobivajte, spreminjajte in brišite podatke z računalniško napravo in določite informacije, ki so shranjene kot podatki.** | Vse informacije, ki jih računalniška naprava shranjuje in obdeluje, se imenujejo podatki. Podatki so lahko slike, besedilni dokumenti, zvočne datoteke, programski programi ali aplikacije, video datoteke itd. Ko študenti uporabljajo programsko opremo za dokončanje nalog na računalniški napravi, bodo manipulirali s podatki. | Podatki in analiza | Skladiščenje | Razvijanje in uporaba abstrakcij | 4.2 |
| 1A-DA-06 | 1A | K-2 | **Zbiranje in predstavitev istih podatkov v različnih vizualnih oblikah.** | Zbiranje in uporaba podatkov o svetu okoli njih je rutinski del življenja in vpliva na življenje ljudi. Študenti bi lahko zbirali podatke o vremenu, kot so sončni dnevi v primerjavi z deževnimi dnevi, temperatura na začetku šolskega dne in konec šolskega dne ali centimetri dežja med nevihto. Študenti bi lahko prešteli število kosov vsake barve sladkarij v vrečki sladkarij, kot so Skittles ali M&Ms. Študenti bi lahko ustvarili ankete o stvareh, ki jih zanimajo, kot so najljubša hrana, hišni ljubljenčki ali televizijske oddaje, in zbrali odgovore na svoje ankete od svojih vrstnikov in drugih. Zbrane podatke lahko nato organiziramo v dve ali več vizualizacij, kot so stolpčni grafikon, tortni grafikon ali piktogram. | Podatki in analiza | Vizualizacija in transformacija zbirke | Komuniciranje o računalništvu, razvijanje in uporaba abstrakcij | 7.1, 4.4 |
| 1A-DA-07 | 1A | K-2 | **Prepoznajte in opišite vzorce v ponazoritvah podatkov, kot so grafikoni ali grafi, da naredite napovedi.** | Podatki se lahko uporabijo za sklepanje ali napovedi o svetu. Študenti bi lahko analizirali graf ali tortni grafikon barv v vrečki sladkarij ali povprečja barv v več vrečah sladkarij, ugotovili vzorce, za katere so barve najbolj in najmanj zastopane, in nato napovedali, katere barve bodo imele največ in najmanj v novi vrečki sladkarij. Učenci bi lahko analizirali grafe temperatur, vzetih na začetku šolskega dne in koncu šolskega dne, ugotovili vzorce, kdaj se temperature dvignejo in padejo, in predvidijo, ali menijo, da se bo temperatura dvignila ali znižala ob določenem času dneva, na podlagi opaženega vzorca. | Podatki in analiza | Sklepanje in modeli | Razvijanje in uporaba abstrakcij | 4.1 |
| 1A-AP-08 | 1A | K-2 | **Modelirajte dnevne procese z ustvarjanjem in sledenjem algoritmom (nabori navodil po korakih) za dokončanje nalog.** | Kompozicija je kombinacija manjših nalog v bolj zapletene naloge. Učenci so lahko ustvarili in sledili algoritmom za pripravo preproste hrane, umivanje zob, pripravo na šolo, sodelovanje pri čiščenju. | Algoritmi in programiranje | Algoritmi | Razvijanje in uporaba abstrakcij | 4.4 |
| 1A-AP-09 | 1A | K-2 | **Oblikujte način, kako programi shranjujejo in manipulirajo s podatki z uporabo številk ali drugih simbolov, ki predstavljajo informacije.** | Informacije v resničnem svetu so lahko predstavljene v računalniških programih. Študenti bi lahko uporabili palce navzgor / navzdol kot predstavitev da / ne, uporabili puščice pri pisanju algoritmov za predstavitev smeri ali kodirali in dekodirali besede s številkami, piktogrami ali drugimi simboli za predstavitev črk ali besed. | Algoritmi in programiranje | Spremenljivke | Razvijanje in uporaba abstrakcij | 4.4 |
| 1A-AP-10 | 1A | K-2 | **Razvijte programe z zaporedji in preprostimi zankami, da izrazite ideje ali rešite problem.** | Programiranje se uporablja kot orodje za ustvarjanje izdelkov, ki odražajo širok spekter interesov. Nadzorne strukture določajo vrstni red, v katerem se izvajajo navodila v programu. Zaporedja so vrstni red navodil v programu. Na primer, če dialog ni pravilno zaporedje pri programiranju preproste animirane zgodbe, zgodba ne bo imela smisla. Če ukazi za programiranje robota niso v pravilnem vrstnem redu, robot ne bo opravil želene naloge. Zanke omogočajo večkratno ponavljanje zaporedja kode. Na primer, v programu za prikaz življenjskega cikla metulja bi lahko zanko kombinirali z ukazi za premikanje, da bi omogočili neprekinjeno, vendar nadzorovano gibanje lika. | Algoritmi in programiranje | Kontrola | Ustvarjanje računalniških artefaktov | 5.2 |
| 1A-AP-11 | 1A | K-2 | **Razčlenite (razčlenite) korake, potrebne za rešitev problema, v natančno zaporedje navodil.** | Razgradnja je dejanje razčlenitve nalog na enostavnejše naloge. Učenci so lahko razčlenili korake, potrebne za izdelavo sendviča z arašidovim maslom in želejem, umivanje zob, risanje oblike, premikanje lika po zaslonu ali reševanje ravni aplikacije za kodiranje. | Algoritmi in programiranje | Modularnost | Prepoznavanje in definiranje računalniških problemov | 3.2 |
| 1A-AP-12 | 1A | K-2 | **Razvijte načrte, ki opisujejo zaporedje dogodkov, ciljev in pričakovanih rezultatov programa.** | Če ustvarite načrt za to, kaj bo program naredil, pojasnite korake, ki bodo potrebni za ustvarjanje programa, in jih lahko uporabite za preverjanje, ali je program pravilen. Študenti bi lahko ustvarili načrtovalni dokument, kot je zemljevid zgodbe, snemalna knjiga ali zaporedni grafični organizator, da bi ponazorili, kaj bo njihov program naredil. Študenti na tej stopnji lahko zaključijo postopek načrtovanja s pomočjo svojih učiteljev. | Algoritmi in programiranje | Razvoj programa | Ustvarjanje računalniških artefaktov, komuniciranje o računalništvu | 5.1, 7.2 |
| 1A-AP-13 | 1A | K-2 | **Navedite pripisovanje pri uporabi idej in stvaritev drugih med razvojem programov.** | Uporaba računalnikov je povezana z določeno stopnjo odgovornosti. Študenti bi morali priznati artefakte, ki so jih ustvarili drugi, kot so slike, glasba in koda. Zasluge se lahko dodelijo ustno, če svoje delo predstavijo razredu, ali pisno ali ustno, če delijo delo na razrednem blogu ali spletni strani. Pravilno pripisovanje na tej stopnji ne zahteva uradnega citiranja, na primer v bibliografiji ali citiranem dokumentu. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.3 |
| 1A-AP-14 | 1A | K-2 | **Odpravljanje napak (prepoznavanje in odpravljanje) napak v algoritmu ali programu, ki vključuje zaporedja in preproste zanke.** | Algoritmi ali programi morda ne bodo vedno delovali pravilno. Študenti bi morali biti sposobni uporabljati različne strategije, kot so spreminjanje zaporedja korakov, sledenje algoritmu po korakih ali poskusi in napake za odpravljanje težav v algoritmih in programih. | Algoritmi in programiranje | Razvoj programa | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.2 |
| 1A-AP-15 | 1A | K-2 | **Z uporabo pravilne terminologije opišite sprejete korake in izbire med iterativnim procesom razvoja programa.** | Na tej stopnji bi morali biti študenti sposobni govoriti ali pisati o ciljih in pričakovanih rezultatih programov, ki jih ustvarijo, in o odločitvah, ki so jih sprejeli pri ustvarjanju programov. To je mogoče storiti z uporabo revij za kodiranje, razprav z učiteljem, predstavitev razreda ali blogov. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.2 |
| 1A-IC-16 | 1A | K-2 | **Primerjajte, kako ljudje živijo in delajo pred in po uvedbi ali sprejetju nove računalniške tehnologije.** | Računalniška tehnologija je pozitivno in negativno spremenila način življenja in dela ljudi. V preteklosti, če so učenci želeli brati o temi, so potrebovali dostop do knjižnice, da bi našli knjigo o njej. Danes si lahko učenci ogledajo in berejo informacije o temi na internetu ali pa lahko prenesejo e-knjige o njej neposredno v napravo. Takšne informacije so lahko na voljo v več kot enem jeziku in jih je mogoče prebrati študentu, kar omogoča veliko dostopnost. | Vplivi računalništva | Kultura | Komuniciranje o računalništvu | 7 |
| 1A-IC-17 | 1A | K-2 | **Sodelujte spoštljivo in odgovorno z drugimi na spletu.** | Spletna komunikacija omogoča pozitivne interakcije, kot je izmenjava idej z mnogimi ljudmi, vendar javna in anonimna narava spletne komunikacije omogoča tudi zastrašujoče in neprimerno vedenje v obliki spletnega ustrahovanja. Študenti bi lahko delili svoje delo na blogih ali v drugih spletnih prostorih za sodelovanje, pri čemer bi pazili, da ne delijo informacij, ki so neprimerne ali bi jih lahko osebno identificirale z drugimi. Študenti bi lahko drugim prijazno in spoštljivo posredovali povratne informacije o svojem delu in bi lahko odraslim povedali, če drugi delijo stvari, ki jih ne bi smeli deliti, ali pa z drugimi ravnajo neprijazno ali nespoštljivo v spletnih prostorih za sodelovanje. | Vplivi računalništva | Družbene interakcije | Sodelovanje na področju računalništva | 2.1 |
| 1A-IC-18 | 1A | K-2 | **Podatki za prijavo naj bodo zasebni in se ustrezno odjavite iz naprav.** | Ljudje uporabljajo računalniško tehnologijo na načine, ki lahko pomagajo ali škodujejo sebi ali drugim. Škodljivo vedenje, kot je deljenje zasebnih podatkov in puščanje javnih naprav prijavljenih, je treba prepoznati in se jim izogibati. | Vplivi računalništva | Varnostno pravo in etika | Komuniciranje o računalništvu | 7.3 |
| 1B-CS-01 | 1B | 3-5 | **Opišite, kako delujejo notranji in zunanji deli računalniških naprav, da tvorijo sistem.** | Računalniške naprave so pogosto odvisne od drugih naprav ali komponent. Na primer, robot je odvisen od fizično pritrjenega svetlobnega senzorja za zaznavanje sprememb svetlosti, medtem ko je svetlobni senzor odvisen od moči robota. Vnos s tipkovnico ali klik miške lahko povzroči dejanje ali prikaz informacij na zaslonu; To se lahko zgodi le zato, ker ima računalnik procesor, ki oceni, kaj se dogaja navzven, in proizvede ustrezne odzive. Študenti morajo opisati, kako naprave in komponente medsebojno delujejo z uporabo pravilne terminologije. | Računalniški sistemi | Naprave | Komuniciranje o računalništvu | 7.2 |
| 1B-CS-02 | 1B | 3-5 | **Modelirajte, kako računalniška strojna in programska oprema delujeta skupaj kot sistem za izpolnjevanje nalog.** | Da bi oseba lahko opravljala naloge z računalnikom, je potrebna strojna in programska oprema. Na tej stopnji mora model vključevati le osnovne elemente računalniškega sistema, kot so vhod, izhod, procesor, senzorji in pomnilnik. Študenti bi lahko narišli model na papir ali v programu za risanje, programirali animacijo, da bi jo prikazali, ali jo prikazali tako, da bi to na nek način izvedli. | Računalniški sistemi | Strojna in programska oprema | Razvijanje in uporaba abstrakcij | 4.4 |
| 1B-CS-03 | 1B | 3-5 | **Določite možne rešitve za reševanje preprostih težav s strojno in programsko opremo z uporabo običajnih strategij za odpravljanje težav.** | Čeprav se računalniški sistemi lahko razlikujejo, lahko na vseh uporabite skupne strategije za odpravljanje težav. Študenti bi morali biti sposobni prepoznati rešitve za težave, kot so naprava, ki se ne odziva, brez napajanja, brez omrežja, zrušitve aplikacije, brez zvoka ali vnosa gesla ne deluje. Če bi se v šoli pojavile napake, bi bil cilj, da bi učenci uporabili različne strategije, kot so ponovni zagon naprave, preverjanje napajanja, preverjanje razpoložljivosti omrežja, zapiranje in ponovno odpiranje aplikacije, preverjanje, ali so zvočniki vklopljeni ali so priključene slušalke, in preverjanje, da tipka Caps Lock ni vklopljena, da bi rešili te težave, kadar je mogoče. | Računalniški sistemi | Odpravljanje težav | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.2 |
| 1B-NI-04 | 1B | 3-5 | **Modelirajte, kako se informacije razčlenijo na manjše dele, se prenašajo kot paketi prek več naprav prek omrežij in interneta ter ponovno sestavljajo na cilju.** | Informacije se pošiljajo in prejemajo po fizičnih ali brezžičnih poteh. Razdeljen je na manjše dele, imenovane paketi, ki se pošljejo neodvisno in ponovno sestavijo na cilju. Študenti bi morali pokazati svoje razumevanje tega pretoka informacij, na primer tako, da narisajo model načina prenosa paketov, programirajo animacijo, ki pokaže, kako se paketi prenašajo, ali pa to pokažejo z odklopljeno dejavnostjo, ki jih na nek način odigra. | Omrežja in internet | Omrežna komunikacija in organizacija | Razvijanje in uporaba abstrakcij | 4.4 |
| 1B-NI-05 | 1B | 3-5 | **Pogovorite se o resničnih težavah s kibernetsko varnostjo in o tem, kako je mogoče zaščititi osebne podatke.** | Tako kot ščitimo svojo osebno lastnino brez povezave, moramo zaščititi tudi svoje naprave in podatke, shranjene v njih. Informacije je mogoče zaščititi z različnimi varnostnimi ukrepi. Ti ukrepi so lahko fizični in/ali digitalni. Študenti bi lahko razpravljali ali uporabili dejavnost dnevnika ali bloga, da ustno ali pisno razložijo teme, ki se nanašajo na osebna vprašanja kibernetske varnosti. Teme razprav bi lahko temeljile na trenutnih dogodkih, povezanih s kibernetsko varnostjo, ali temah, ki so uporabne za študente, kot so potreba po varnostnem kopiranju podatkov za zaščito pred izgubo, kako ustvariti močna gesla in pomen nedeljenja gesel ali zakaj bi morali namestiti in posodabljati protivirusno programsko opremo za zaščito podatkov in sistemov. | Omrežja in internet | Kibernetska varnost | Prepoznavanje in definiranje računalniških problemov | 3.1 |
| 1B-DA-06 | 1B | 3-5 | **Vizualno organizirajte in predstavite zbrane podatke, da poudarite odnose in podprete zahtevek.** | Surovi podatki sami po sebi nimajo velikega pomena. Podatki so pogosto razvrščeni ali združeni, da se zagotovi dodatna jasnost. Organiziranje podatkov lahko olajša razlago in sporočanje drugim. Podatkovne točke se lahko združijo s številnimi skupnimi značilnostmi. Z istimi podatki bi lahko manipulirali na različne načine, da bi poudarili določene vidike ali dele nabora podatkov. Na primer, nabor podatkov o športnih ekipah je mogoče razvrstiti po zmagah, doseženih točkah ali dovoljenih točkah, nabor podatkov o vremenu pa po visokih temperaturah, nizkih temperaturah ali padavinah. | Podatki in analiza | Vizualizacija in transformacija zbirke | Komuniciranje o računalništvu | 7.1 |
| 1B-DA-07 | 1B | 3-5 | **Uporabite podatke za poudarjanje ali predlaganje vzročno-posledičnih povezav, napovedovanje izidov ali sporočanje ideje.** | Natančnost analize podatkov je povezana s tem, kako realno so podatki predstavljeni. Sklepanje ali napovedi, ki temeljijo na podatkih, so manj verjetno, da bodo točni, če podatki niso zadostni ali če so podatki na kakršen koli način napačni. Študenti bi morali biti sposobni sklicevati se na podatke, ko sporočajo idejo. Na primer, da bi raziskali razmerje med hitrostjo, časom in razdaljo, bi lahko študenti upravljali robota z enakomerno hitrostjo in v vse večjih časovnih intervalih, da bi predvideli, kako daleč robot potuje s to hitrostjo. Da bi naredili natančno napoved, en ali dva poskusa različnih časov ne bi bila dovolj. Robot lahko zbira tudi podatke o temperaturi iz senzorja, vendar ti podatki ne bi bili pomembni za nalogo. Študenti morajo opraviti tudi natančne meritve razdalje, ki jo robot prepotuje, da bi razvili veljavno napoved. Študenti bi lahko vsak dan zabeležili temperaturo opoldne kot podlago, da bi pokazali, da so temperature v določenih mesecih leta višje. Če se temperature ne zabeležijo ob nešolskih dneh ali so zabeležene napačno ali ob različnih časih dneva, bi bili podatki nepopolni in ideje, ki se sporočajo, bi lahko bile netočne. Študenti lahko zabeležijo tudi dan v tednu, ko so bili podatki zbrani, vendar to ne bi imelo pomena za to, ali so temperature višje ali nižje. Da bi imeli dovolj in natančnih podatkov, o katerih bi lahko sporočili idejo, bodo študenti morda želeli uporabiti podatke, ki jih zagotavlja vladna vremenska agencija. | Podatki in analiza | Sklepanje in modeli | Komuniciranje o računalništvu | 7.1 |
| 1B-AP-08 | 1B | 3-5 | **Primerjajte in izboljšajte več algoritmov za isto nalogo in določite, kateri je najprimernejši.** | Različni algoritmi lahko dosežejo enak rezultat, čeprav je včasih en algoritem lahko najprimernejši za določeno situacijo. Študenti bi morali biti sposobni preučiti različne načine reševanja iste naloge in se odločiti, katera bi bila najboljša rešitev. Študenti bi lahko na primer uporabili zemljevid in načrtovali več algoritmov, da bi prišli iz ene točke v drugo. Lahko bi si ogledali poti, ki jih predlaga programska oprema za kartiranje, in spremenili pot v nekaj, kar bi bilo boljše, glede na to, katera pot je najkrajša ali najhitrejša ali bi se izognila težavam. Učenci lahko primerjajo algoritme, ki opisujejo, kako se pripraviti na šolo. Drug primer bi lahko bil pisanje različnih algoritmov za risanje pravilnega mnogokotnika in določitev, kateri algoritem bi bilo najlažje spremeniti ali spremeniti za risanje drugega mnogokotnika. | Algoritmi in programiranje | Algoritmi | Preizkušanje in izpopolnjevanje računalniških artefaktov, prepoznavanje in definiranje računskih problemov | 6.3, 3.3 |
| 1B-AP-09 | 1B | 3-5 | **Ustvarite programe, ki uporabljajo spremenljivke za shranjevanje in spreminjanje podatkov.** | Spremenljivke se uporabljajo za shranjevanje in spreminjanje podatkov. Na tej ravni zadostuje razumevanje, kako uporabljati spremenljivke. Na primer, učenci lahko uporabijo matematične operacije, da dodajo rezultat igre ali odštejejo od števila življenj, ki so na voljo v igri. Uporaba spremenljivke kot odštevalnika je še en primer. | Algoritmi in programiranje | Spremenljivke | Ustvarjanje računalniških artefaktov | 5.2 |
| 1B-AP-10 | 1B | 3-5 | **Ustvarite programe, ki vključujejo zaporedja, dogodke, zanke in pogoje.** | Nadzorne strukture določajo vrstni red (zaporedje), v katerem se navodila izvajajo v programu in jih je mogoče kombinirati za podporo ustvarjanju bolj zapletenih programov. Dogodki omogočajo, da se deli programa izvajajo na podlagi določenega dejanja. Na primer, učenci bi lahko napisali program za razlago vodnega cikla in ko kliknete določeno komponento (dogodek), bi program prikazal informacije o tem delu vodnega cikla. Pogoji omogočajo izvedbo dela kode v programu, ko je določen pogoj resničen. Na primer, učenci bi lahko napisali matematično igro, ki postavlja vprašanja o množenju in nato uporabi pogoj, da preveri, ali je vneseni odgovor pravilen ali ne. Zanke omogočajo večkratno ponavljanje zaporedja kode. Na primer, v programu, ki proizvaja animacijo o slavnem zgodovinskem liku, bi lahko učenci uporabili zanko, da bi lik hodil po zaslonu, ko se predstavijo. | Algoritmi in programiranje | Kontrola | Ustvarjanje računalniških artefaktov | 5.2 |
| 1B-AP-11 | 1B | 3-5 | **Razčlenite (razčlenite) težave na manjše, obvladljive podprobleme, da olajšate proces razvoja programa.** | Razgradnja je dejanje razčlenitve nalog na enostavnejše naloge. Učenci bi lahko na primer ustvarili animacijo tako, da zgodbo ločili na različne prizore. Za vsak prizor bi izbrali ozadje, postavili like in programirali dejanja. | Algoritmi in programiranje | Modularnost | Prepoznavanje in definiranje računalniških problemov | 3.2 |
| 1B-AP-12 | 1B | 3-5 | **Spremenite, remiksirajte ali vključite dele obstoječega programa v svoje delo, da razvijete nekaj novega ali dodate naprednejše funkcije.** | Programi se lahko razdelijo na manjše dele, ki jih je mogoče vključiti v nove ali obstoječe programe. Študenti bi lahko na primer spremenili vnaprej napisano kodo iz igre za enega igralca, da bi ustvarili igro za dva igralca z nekoliko drugačnimi pravili, remiksirali in dodali še en prizor animirani zgodbi, uporabili kodo, da bi žoga odskočila od drugega programa v novi košarkarski igri ali spremenili sliko, ki jo je ustvaril drug učenec. | Algoritmi in programiranje | Modularnost | Ustvarjanje računalniških artefaktov | 5.3 |
| 1B-AP-13 | 1B | 3-5 | **Uporabite iterativni postopek za načrtovanje razvoja programa z vključitvijo perspektiv drugih in upoštevanjem uporabniških preferenc.** | Načrtovanje je pomemben del iterativnega procesa razvoja programa. Študenti opisujejo ključne značilnosti, časovne omejitve in omejitve virov ter pričakovanja uporabnikov. Študenti morajo načrt dokumentirati kot na primer snemalno knjigo, diagram poteka, psevdokodo ali zemljevid zgodbe. | Algoritmi in programiranje | Razvoj programa | Spodbujanje vključujoče računalniške kulture, ustvarjanje računalniških artefaktov | 1.1, 5.1 |
| 1B-AP-14 | 1B | 3-5 | **Upoštevajte pravice intelektualne lastnine in navedite ustrezno avtorstvo pri ustvarjanju ali remiksiranju programov.** | Pravice intelektualne lastnine se lahko razlikujejo glede na državo, vendar zakoni o avtorskih pravicah avtorju dela dajejo nabor pravic, ki drugim preprečujejo kopiranje dela in njegovo uporabo na načine, ki jim morda niso všeč. Študenti bi morali prepoznati primere remiksiranja, ko so ideje izposojene in ponovljene, ter pripisati prvotnemu ustvarjalcu. Študenti bi morali razmisliti tudi o skupnih licencah, ki omejujejo ali omejujejo uporabo računalniških artefaktov, kot so slike in glasba, prenesena z interneta. Na tej stopnji mora biti avtorstvo napisano v obliki, ki jo zahteva učitelj, in mora biti vedno vključeno v vse programe, ki so v skupni rabi na spletu. | Algoritmi in programiranje | Razvoj programa | Ustvarjanje računalniških artefaktov, komuniciranje o računalništvu | 5.2, 7.3 |
| 1B-AP-15 | 1B | 3-5 | **Preizkusite in razhroščite (prepoznajte in popravite napake) programa ali algoritma, da zagotovite, da deluje tako, kot je predvideno.** | Ko študenti razvijajo programe, bi morali nenehno preizkušati te programe, da bi videli, da delajo, kar je bilo pričakovano, in popravijo (odpravljanje napak), morebitne napake. Študenti bi morali biti sposobni tudi uspešno odpravljati napake v programih, ki so jih ustvarili drugi. | Algoritmi in programiranje | Razvoj programa | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.1, 6.2 |
| 1B-AP-16 | 1B | 3-5 | **Prevzemite različne vloge z vodstvom učiteljev pri sodelovanju z vrstniki med fazami oblikovanja, izvajanja in pregleda razvoja programa.** | Sodelovalno računalništvo je postopek izvajanja računalniške naloge z delom v parih ali v skupinah. Ker vključuje prošnjo za prispevke in povratne informacije drugih, lahko učinkovito sodelovanje privede do boljših rezultatov kot samostojno delo. Študenti bi se morali med razvojem programa izmenjevati v različnih vlogah, kot so zapisovalec, moderator, preizkuševalec programov ali "gonilnik" računalnika. | Algoritmi in programiranje | Razvoj programa | Sodelovanje na področju računalništva | 2.2 |
| 1B-AP-17 | 1B | 3-5 | **Opišite izbire med razvojem programa s komentarji kode, predstavitvami in predstavitvami.** | Ljudje komunicirajo o svoji kodi, da bi drugim pomagali razumeti in uporabljati njihove programe. Drug namen sporočanja svojih oblikovalskih odločitev je pokazati razumevanje svojega dela. Te razlage se lahko manifestirajo kot komentarji kode v vrstici za sodelavce in ocenjevalce ali kot del povzetke predstavitve, kot je sprehod po kodi ali dnevnik kodiranja. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.2 |
| 1B-IC-18 | 1B | 3-5 | **Razpravljajte o računalniških tehnologijah, ki so spremenile svet, in izrazite, kako te tehnologije vplivajo na kulturne prakse.** | Ustvarja se nova računalniška tehnologija in spreminjajo obstoječe tehnologije iz več razlogov, vključno s povečanjem njihovih koristi, zmanjšanjem tveganj in izpolnjevanjem družbenih potreb. Učenci bi morali pod vodstvom učitelja razpravljati o temah, ki se nanašajo na zgodovino tehnologije in spremembe v svetu zaradi tehnologije. Teme bi lahko temeljile na aktualnih novicah, kot so robotika, brezžični internet, mobilne računalniške naprave, sistemi GPS, nosljivo računalništvo ali kako so družbeni mediji vplivali na družbene in politične spremembe. | Vplivi računalništva | Kultura | Prepoznavanje in definiranje računalniških problemov | 3.1 |
| 1B-IC-19 | 1B | 3-5 | **Razmišljajte o načinih za izboljšanje dostopnosti in uporabnosti tehnoloških izdelkov za različne potrebe in želje uporabnikov.** | Razvoj in spreminjanje računalniške tehnologije poganjajo potrebe in želje ljudi in lahko različno vplivajo na skupine. Predvidevanje potreb in želja različnih končnih uporabnikov zahteva, da študenti namenoma upoštevajo potencialne perspektive uporabnikov z različnimi ozadji, stopnjami sposobnosti, stališči in invalidnostmi. Na primer, učenci lahko razmislijo o uporabi govora in besedila, ko želijo prenesti informacije v igri. Morda bodo želeli tudi spreminjati vrste programov, ki jih ustvarjajo, vedoč, da vsi ne delijo svojega okusa. | Vplivi računalništva | Kultura | Spodbujanje vključujoče računalniške kulture | 1.2 |
| 1B-IC-20 | 1B | 3-5 | **Poiščite različne perspektive za izboljšanje računalniških artefaktov.** | Računalništvo omogoča sodelovanje in izmenjavo idej ter omogoča različne perspektive. Učenci lahko na primer poiščejo povratne informacije od drugih skupin v svojem razredu ali učencev na drugi stopnji. Ali pa lahko pod vodstvom učitelja uporabijo orodja za videokonference ali druge spletne prostore za sodelovanje, kot so blogi, wikiji, forumi ali komentarji na spletnih straneh, da zberejo povratne informacije posameznikov in skupin o programskih projektih. | Vplivi računalništva | Družbene interakcije | Spodbujanje vključujoče računalniške kulture | 1.1 |
| 1B-IC-21 | 1B | 3-5 | **Uporabljajte medije v javni domeni ali Creative Commons in se vzdržite kopiranja ali uporabe gradiva, ki so ga ustvarili drugi, brez dovoljenja.** | Etični zapleti izhajajo iz priložnosti, ki jih ponuja računalništvo. Enostavnost pošiljanja in prejemanja kopij medijev v internetu, kot so videoposnetki, fotografije in glasba, ustvarja priložnost za nepooblaščeno uporabo, kot je spletno piratstvo, in neupoštevanje avtorskih pravic. Študenti bi morali razmisliti o licencah za računalniške artefakte, ki jih želijo uporabiti. Licenca za preneseno slikovno ali zvočno datoteko ima lahko na primer omejitve, ki prepovedujejo spreminjanje, zahtevajo navedbo avtorstva ali v celoti prepovedujejo uporabo. | Vplivi računalništva | Varnostno pravo in etika | Komuniciranje o računalništvu | 7.3 |
| 2-CS-01 | 2 | 6-8 | **Priporočite izboljšave zasnove računalniških naprav na podlagi analize interakcije uporabnikov z napravami.** | Študija interakcije med človekom in računalnikom (HCI) lahko izboljša zasnovo naprav, vključno s strojno in programsko opremo. Študenti morajo dati priporočila za obstoječe naprave (npr. Prenosni računalnik, telefon ali tablični računalnik) ali oblikovati lastne komponente ali vmesnik (npr. ustvariti lastne krmilnike). Učitelji lahko učence vodijo, da razmislijo o uporabnosti skozi več leč, vključno z dostopnostjo, ergonomijo in učitostjo. Pripomočki za pomoč uporabnikom s posebnimi potrebami na primer ponujajo zmogljivosti, kot so optično branje pisnih informacij in njihovo pretvorbo v govor. | Računalniški sistemi | Naprave | Prepoznavanje in definiranje računalniških problemov | 3.3 |
| 2-CS-02 | 2 | 6-8 | **Oblikujte projekte, ki združujejo komponente strojne in programske opreme za zbiranje in izmenjavo podatkov.** | Zbiranje in izmenjava podatkov vključuje vnos, izhod, shranjevanje in obdelavo. Kadar je mogoče, morajo študenti izbrati komponente strojne in programske opreme za svoje projektne načrte ob upoštevanju dejavnikov, kot so funkcionalnost, stroški, velikost, hitrost, dostopnost in estetika. Komponente za mobilno aplikacijo lahko na primer vključujejo merilnik pospeška, GPS in prepoznavanje govora. Izbira naprave, ki se brezžično poveže prek povezave Bluetooth v primerjavi s fizično povezavo USB, vključuje kompromis med mobilnostjo in potrebo po dodatnem viru napajanja za brezžično napravo. | Računalniški sistemi | Strojna in programska oprema | Ustvarjanje računalniških artefaktov | 5.1 |
| 2-CS-03 | 2 | 6-8 | **Sistematično prepoznajte in odpravljajte težave z računalniškimi napravami in njihovimi komponentami.** | Ker lahko računalniška naprava komunicira z medsebojno povezanimi napravami v sistemu, težave morda niso posledica same računalniške naprave, temveč naprav, ki so z njo povezane. Tako kot piloti uporabljajo kontrolne sezname za odpravljanje težav z letalskimi sistemi, bi morali študenti uporabiti podoben, strukturiran postopek za odpravljanje težav z računalniškimi sistemi in zagotoviti, da potencialne rešitve ne bodo spregledane. Primeri strategij za odpravljanje težav vključujejo sledenje diagramu poteka odpravljanja težav, spreminjanje programske opreme, da preverite, ali bo strojna oprema delovala, preverjanje povezav in nastavitev ter zamenjavo delujočih komponent. | Računalniški sistemi | Odpravljanje težav | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.2 |
| 2-NI-04 | 2 | 6-8 | **Modelirajte vlogo protokolov pri prenosu podatkov prek omrežij in interneta.** | Protokoli so pravila, ki določajo, kako se pošiljajo sporočila med računalniki. Določajo, kako hitro in varno se informacije prenašajo prek omrežij in interneta ter kako ravnati z napakami pri prenosu. Študenti bi morali modelirati, kako se podatki pošiljajo s protokoli, da izberejo najhitrejšo pot, obravnavajo manjkajoče informacije in varno dostavijo občutljive podatke. Učenci bi lahko na primer oblikovali načrt za ponovno pošiljanje izgubljenih informacij ali za razlago slike, ki ima manjkajoče dele. Prednostna naloga na tej ravni je razumevanje namena protokolov in kako omogočajo varno in brezhibno komunikacijo. Poznavanje podrobnosti o tem, kako delujejo določeni protokoli, ni pričakovati. | Omrežja in internet | Omrežna komunikacija in organizacija | Razvijanje in uporaba abstrakcij | 4.4 |
| 2-NI-05 | 2 | 6-8 | **Pojasnite, kako fizični in digitalni varnostni ukrepi ščitijo elektronske informacije.** | Podatki, ki so shranjeni v spletu, so ranljivi za neželen dostop. Primeri fizičnih varnostnih ukrepov za zaščito podatkov vključujejo skrivanje gesel, zaklepanje vrat, izdelavo varnostnih kopij na zunanjih napravah za shranjevanje in brisanje naprave za shranjevanje pred ponovno uporabo. Primeri digitalnih varnostnih ukrepov vključujejo varna skrbniška gesla usmerjevalnika, požarne zidove, ki omejujejo dostop do zasebnih omrežij, in uporabo protokola, kot je HTTPS, za zagotovitev varnega prenosa podatkov. | Omrežja in internet | Kibernetska varnost | Komuniciranje o računalništvu | 7.2 |
| 2-NI-06 | 2 | 6-8 | **Uporabite več metod šifriranja za modeliranje varnega prenosa informacij.** | Šifriranje je lahko tako preprosto kot zamenjava črk ali tako zapleteno kot sodobne metode, ki se uporabljajo za zaščito omrežij in interneta. Študenti bi morali kodirati in dekodirati sporočila z različnimi metodami šifriranja in razumeti različne stopnje kompleksnosti, ki se uporabljajo za skrivanje ali varovanje informacij. Učenci bi lahko na primer zavarovali sporočila z metodami, kot so Cezarjeve šifre ali steganografija (tj. Skrivanje sporočil znotraj slike ali drugih podatkov). Prav tako lahko modelirajo bolj zapletene metode, kot je šifriranje javnega ključa, z dejavnostmi, ki niso priključene. | Omrežja in internet | Kibernetska varnost | Razvijanje in uporaba abstrakcij | 4.4 |
| 2-DA-07 | 2 | 6-8 | **Predstavite podatke z več shemami kodiranja.** | Predstavitve podatkov se pojavljajo na več ravneh abstrakcije, od fizičnega shranjevanja bitov do razporeditve informacij v organizirane oblike (npr. tabele). Študenti morajo predstavljati iste podatke na več načinov. Na primer, učenci lahko predstavljajo isto barvo z uporabo binarnih, RGB vrednosti, šestnajstih kod (predstavitve na nizki ravni), pa tudi oblike, ki jih ljudje razumejo, vključno z besedami, simboli in digitalnimi prikazi barv (predstavitve na visoki ravni). | Podatki in analiza | Skladiščenje | Razvijanje in uporaba abstrakcij | 4 |
| 2-DA-08 | 2 | 6-8 | **Zbirajte podatke z računalniškimi orodji in jih preoblikujte tako, da bodo bolj uporabni in zanesljivi.** | Ker študenti še naprej gradijo na svoji sposobnosti organiziranja in vizualne predstavitve podatkov, da bi podprli trditev, bodo morali razumeti, kdaj in kako preoblikovati podatke v ta namen. Študenti bi morali preoblikovati podatke, da bi odstranili napake, poudarili ali izpostavili odnose in / ali olajšali obdelavo računalnikov. Čiščenje podatkov je pomembna preobrazba za zagotavljanje dosledne oblike in zmanjšanje hrupa in napak (npr. odstranjevanje nepomembnih odgovorov v anketi). Primer transformacije, ki poudarja odnos, predstavlja moške in ženske kot odstotke celote in ne kot posamezno štetje. | Podatki in analiza | Vizualizacija in transformacija zbirke | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.3 |
| 2-DA-09 | 2 | 6-8 | **Izboljšajte računalniške modele na podlagi podatkov, ki so jih ustvarili.** | Model je lahko programirana simulacija dogodkov ali predstavitev tega, kako so različni podatki povezani. Da bi izboljšali model, morajo učenci razmisliti, katere podatkovne točke so pomembne, kako so podatkovne točke med seboj povezane in ali so podatki točni. Študenti lahko na primer napovedujejo, kako daleč bo žoga potovala na podlagi tabele podatkov, povezanih z višino in kotom proge. Učenci bi lahko nato preizkusili in izboljšali svoj model s primerjavo predvidenih in dejanskih rezultatov ter razmislili, ali so pomembni drugi dejavniki (npr. Velikost in masa žoge). Poleg tega bi študenti lahko izboljšali mehaniko igre na podlagi rezultatov testov, da bi bila igra bolj uravnotežena ali poštena. | Podatki in analiza | Sklepanje in modeli | Ustvarjanje računalniških artefaktov, razvoj in uporaba abstrakcij | 5.3, 4.4 |
| 2-AP-10 | 2 | 6-8 | **Uporabite diagrame poteka in / ali psevdokodo za reševanje zapletenih problemov kot algoritmov.** | Kompleksni problemi so problemi, ki bi jih študenti težko rešili računsko. Študenti bi morali uporabiti psevdokodo in / ali diagrame poteka za organizacijo in zaporedje algoritma, ki obravnava zapleten problem, čeprav morda dejansko ne programirajo rešitev. Študenti lahko na primer izrazijo algoritem, ki ustvari priporočilo za nakup superg na podlagi vnosov, kot so velikost, barve, blagovna znamka, udobje in stroški. Testiranje algoritma s širokim naborom vnosov in uporabnikov omogoča študentom, da izboljšajo svoj algoritem priporočil in identificirajo druge vnose, ki so jih morda prvotno izključili. | Algoritmi in programiranje | Algoritmi | Razvijanje in uporaba abstrakcij | 4.4, 4.1 |
| 2-AP-11 | 2 | 6-8 | **Ustvarite jasno poimenovane spremenljivke, ki predstavljajo različne podatkovne tipe, in izvedite operacije z njihovimi vrednostmi.** | Spremenljivka je kot vsebnik z imenom, v katerem se lahko vsebina spremeni, ime (identifikator) pa ne. Pri načrtovanju in razvoju programov se morajo študenti odločiti, kdaj in kako bodo razglasili in poimenovali nove spremenljivke. Študenti bi morali uporabljati konvencije o poimenovanju za izboljšanje berljivosti programa. Primeri operacij vključujejo dodajanje točk v rezultat, združevanje uporabniškega vnosa z besedami za ustvarjanje stavka, spreminjanje velikosti slike ali dodajanje imena na seznam ljudi. | Algoritmi in programiranje | Spremenljivke | Ustvarjanje računalniških artefaktov | 5.1, 5.2 |
| 2-AP-12 | 2 | 6-8 | **Oblikujte in iterativno razvijajte programe, ki združujejo nadzorne strukture, vključno z ugnezdenimi zankami in sestavljenimi pogoji.** | Nadzorne strukture je mogoče kombinirati na več načinov. Ugnezdene zanke so zanke, ki so nameščene v zanke. Sestavljeni pogojniki združujejo dva ali več pogojev v logično razmerje (npr. z uporabo AND, OR in NOT), gnezdenje pogojev drug v drugega pa omogoča, da rezultat enega pogoja vodi do drugega. Na primer, pri programiranju interaktivne zgodbe bi lahko učenci uporabili sestavljeni pogoj v zanki za odklepanje vrat le, če ima lik ključ IN se dotakne vrat. | Algoritmi in programiranje | Kontrola | Ustvarjanje računalniških artefaktov | 5.1, 5.2 |
| 2-AP-13 | 2 | 6-8 | **Razčlenite težave in podprobleme na dele, da olajšate načrtovanje, izvajanje in pregled programov.** | Študenti bi morali razčleniti probleme na podprobleme, ki jih je mogoče nadalje razdeliti na manjše dele. Dekompozicija olajša vidike razvoja programa, tako da študentom omogoča, da se osredotočijo na en kos naenkrat (npr. Pridobivanje prispevkov od uporabnika, obdelava podatkov in prikaz rezultata uporabniku). Razgradnja omogoča tudi različnim študentom, da delajo na različnih delih hkrati. Animacije se lahko na primer razčlenijo na več prizorov, ki jih je mogoče razviti neodvisno. | Algoritmi in programiranje | Modularnost | Prepoznavanje in definiranje računalniških problemov | 3.2 |
| 2-AP-14 | 2 | 6-8 | **Ustvarite postopke s parametri za organiziranje kode in lažjo ponovno uporabo.** | Študenti morajo ustvariti postopke in / ali funkcije, ki se uporabljajo večkrat v programu za ponavljanje skupin navodil. Te postopke je mogoče posplošiti z določitvijo parametrov, ki ustvarjajo različne izhode za širok spekter vhodov. Na primer, postopek za risanje kroga vključuje veliko navodil, vendar jih je mogoče priklicati z enim navodilom, kot je »drawCircle«. Z dodajanjem parametra polmera lahko uporabnik enostavno nariše kroge različnih velikosti. | Algoritmi in programiranje | Modularnost | Razvijanje in uporaba abstrakcij | 4.1, 4.3 |
| 2-AP-15 | 2 | 6-8 | **Poiščite in vključite povratne informacije članov skupine in uporabnikov, da izboljšate rešitev, ki ustreza potrebam uporabnikov.** | Razvojne skupine, ki uporabljajo oblikovanje, osredotočeno na uporabnika, ustvarjajo rešitve (npr. programe in naprave), ki imajo lahko velik družbeni vpliv, kot je aplikacija, ki ljudem z govornimi težavami omogoča, da težko razumljivo izgovorjavo prevedejo v razumljiv jezik. Študenti bi morali začeti iskati različne perspektive v celotnem procesu oblikovanja, da bi izboljšali svoje računalniške artefakte. Premisleki končnega uporabnika lahko vključujejo uporabnost, dostopnost, starosti primerno vsebino, spoštljiv jezik, uporabniško perspektivo, uporabo zaimkov, barvni kontrast in enostavnost uporabe. | Algoritmi in programiranje | Razvoj programa | Sodelovanje na področju računalništva, spodbujanje vključujoče računalniške kulture | 2.3, 1.1 |
| 2-AP-16 | 2 | 6-8 | **Vključite obstoječo kodo, medije in knjižnice v izvirne programe in navedite priznanje avtorstva.** | Gradnja na delu drugih omogoča študentom, da ustvarijo bolj zanimive in močnejše stvaritve. Študenti bi morali uporabljati dele kode, algoritmov in / ali digitalnih medijev v svojih programih in spletnih straneh. Na tej ravni lahko tudi uvozijo knjižnice in se povežejo z vmesniki spletnih aplikacij (API). Na primer, pri ustvarjanju igre s stranskim drsenjem lahko učenci vključijo dele kode, ki ustvarijo realistično gibanje skoka iz igre druge osebe, lahko pa tudi uvozijo slike z licenco Creative Commons, ki jih lahko uporabijo v ozadju. Učenci naj pripišejo avtorstvo izvirnim ustvarjalcem, da priznajo njihove prispevke. | Algoritmi in programiranje | Razvoj programa | Razvijanje in uporaba abstrakcij, ustvarjanje računalniških artefaktov, komuniciranje o računalništvu | 4.2, 5.2, 7.3 |
| 2-AP-17 | 2 | 6-8 | **Sistematično preizkušajte in izpopolnjujte programe z uporabo različnih testnih primerov.** | Primeri uporabe in testni primeri so ustvarjeni in analizirani, da bi bolje zadovoljili potrebe uporabnikov in ocenili, ali programi delujejo, kot je predvideno. Na tej ravni bi moralo testiranje postati premišljen proces, ki je bolj iterativen, sistematičen in proaktiven kot na nižjih ravneh. Študenti bi morali začeti preizkušati programe z upoštevanjem morebitnih napak, na primer, kaj se bo zgodilo, če uporabnik vnese neveljaven vnos (npr. Negativna števila in 0 namesto pozitivnih števil). | Algoritmi in programiranje | Razvoj programa | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.1 |
| 2-AP-18 | 2 | 6-8 | **Razdelite opravila in vzdržujte časovnico projekta pri skupnem razvoju računalniških artefaktov.** | Sodelovanje je pogosta in ključna praksa pri razvoju programiranja. Pogosto veliko posameznikov in skupin skupaj dela na medsebojno odvisnih delih projekta. Študenti bi morali prevzeti vnaprej določene vloge v svojih skupinah in upravljati potek dela projekta z uporabo strukturiranih časovnic. Z vodstvom učiteljev bodo začeli ustvarjati skupne cilje, pričakovanja in pravične delovne obremenitve. Študenti lahko na primer fazo oblikovanja igre razdelijo na načrtovanje snemalne knjige, diagrama poteka in različnih delov mehanike igre. Nato lahko razdelijo naloge in vloge med člane skupine in dodelijo roke. | Algoritmi in programiranje | Razvoj programa | Sodelovanje na področju računalništva | 2.2 |
| 2-AP-19 | 2 | 6-8 | **Dokumentirajte programe, da jih lažje spremljate, preizkušate in odpravljate napake.** | Dokumentacija omogoča ustvarjalcem in drugim, da lažje uporabljajo in razumejo program. Študenti morajo končnim uporabnikom zagotoviti dokumentacijo, ki pojasnjuje njihove artefakte in njihovo delovanje. Študenti lahko na primer zagotovijo pregled projekta in jasna navodila za uporabnike. Prav tako morajo vključiti komentarje v svoj izdelek in sporočiti svoj proces z uporabo oblikovalskih dokumentov, diagramov poteka in predstavitev. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.2 |
| 2-IC-20 | 2 | 6-8 | **Primerjajte kompromise, povezane z računalniškimi tehnologijami, ki vplivajo na vsakodnevne dejavnosti in karierne možnosti ljudi.** | Napredek v računalniški tehnologiji ni niti povsem pozitiven niti negativen. Vendar pa imajo načini, kako ljudje uporabljajo računalniške tehnologije, kompromise. Študenti bi morali upoštevati trenutne dogodke, povezane s širšimi idejami, vključno z zasebnostjo, komunikacijo in avtomatizacijo. Na primer, avtomobili brez voznika lahko povečajo udobje in zmanjšajo število nesreč, vendar so tudi dovzetni za vdor. Nastajajoča industrija bo zmanjšala število voznikov taksijev in skupnih voženj, vendar bo ustvarila več delovnih mest v programskem inženiringu in kibernetski varnosti. | Vplivi računalništva | Kultura | Komuniciranje o računalništvu | 7.2 |
| 2-IC-21 | 2 | 6-8 | **Razpravljajte o vprašanjih pristranskosti in dostopnosti pri oblikovanju obstoječih tehnologij.** | Učenci bi morali preizkusiti in razpravljati o uporabnosti različnih tehnoloških orodij (npr. aplikacij, iger in naprav) z učiteljevim vodstvom. Na primer, programska oprema za prepoznavanje obraza, ki bolje deluje za svetlejše odtenke kože, je bila verjetno razvita s homogeno testno skupino in bi jo bilo mogoče izboljšati z vzorčenjem bolj raznolike populacije. Pri razpravi o dostopnosti lahko študenti opazijo, da omogočanje uporabniku, da spremeni velikost in barve pisave, ne bo le naredilo vmesnika uporabnega za slabovidne ljudi, temveč bo koristilo tudi uporabnikom v različnih situacijah, na primer pri močni dnevni svetlobi ali temni sobi. | Vplivi računalništva | Kultura | Spodbujanje vključujoče računalniške kulture | 1.2 |
| 2-IC-22 | 2 | 6-8 | **Sodelujte s številnimi sodelavci s strategijami, kot so množičenje ali ankete pri ustvarjanju računalniškega artefakta.** | Množičenje je zbiranje storitev, idej ali vsebin velike skupine ljudi, zlasti iz spletne skupnosti. To je mogoče storiti na lokalni ravni (npr. Učilnica ali šola) ali globalni ravni (npr. Starosti primerne spletne skupnosti, kot sta Scratch in Minecraft). Na primer, skupina študentov bi lahko združila animacije, da bi ustvarila mozaik digitalne skupnosti. Prav tako bi lahko pridobili povratne informacije od številnih ljudi z uporabo spletnih skupnosti in elektronskih anket. | Vplivi računalništva | Družbene interakcije | Sodelovanje pri računalništvu, ustvarjanje računalniških artefaktov | 2.4, 5.2 |
| 2-IC-23 | 2 | 6-8 | **Opišite kompromise med omogočanjem, da so informacije javne, in ohranjanjem zasebnosti in varnosti informacij.** | Skupna raba informacij na spletu lahko pomaga vzpostaviti, vzdrževati in krepiti povezave med ljudmi. Umetnikom in oblikovalcem na primer omogoča, da pokažejo svoje talente in dosežejo široko občinstvo. Vendar pa se varnostni napadi pogosto začnejo z osebnimi podatki, ki so javno dostopni na spletu. Socialni inženiring temelji na zavajanju ljudi, da razkrijejo občutljive informacije, in ga je mogoče preprečiti s previdnostjo pred napadi, kot sta lažno predstavljanje in lažno predstavljanje. | Vplivi računalništva | Varnostno pravo in etika | Komuniciranje o računalništvu | 7.2 |
| 3A-CS-01 | 3A | 9-10 | **Pojasnite, kako abstrakcije skrivajo osnovne izvedbene podrobnosti računalniških sistemov, vgrajenih v vsakdanje predmete.** | Računalniške naprave so pogosto integrirane z drugimi sistemi, vključno z biološkimi, mehanskimi in družbenimi sistemi. Medicinski pripomoček je lahko vgrajen v osebo za spremljanje in uravnavanje njenega zdravja, slušni aparat (vrsta pomožne naprave) lahko filtrira določene frekvence in poveča druge, nadzorna naprava, nameščena v motornem vozilu, lahko sledi voznim vzorcem in navadam osebe, naprava za prepoznavanje obraza pa se lahko vgradi v varnostni sistem za identifikacijo osebe. Ustvarjanje integriranih ali vgrajenih sistemov na tej ravni ni pričakovanje. Študenti lahko izberejo vgrajeno napravo, kot je avtomobilski stereo, prepoznajo vrste podatkov (prednastavitve radijske postaje, raven glasnosti) in postopke (povečanje glasnosti, shranjevanje / odpoklic shranjene postaje, izklop zvoka), ki jih vključuje, in razložijo, kako so podrobnosti o izvajanju skrite pred uporabnikom. | Računalniški sistemi | Naprave | Razvijanje in uporaba abstrakcij | 4.1 |
| 3A-CS-02 | 3A | 9-10 | **Primerjajte ravni abstrakcije in interakcije med aplikacijsko programsko opremo, sistemsko programsko opremo in plastmi strojne opreme.** | Na svoji najosnovnejši ravni je računalnik sestavljen iz fizične strojne opreme in električnih impulzov. Več plasti programske opreme je zgrajenih na strojni opremi in komunicira s plastmi nad in pod njimi, da se zmanjša kompleksnost. Sistemska programska oprema upravlja vire računalniške naprave, tako da lahko programska oprema komunicira s strojno opremo. Programska oprema za urejanje besedila na primer komunicira z operacijskim sistemom, da prejme vnos s tipkovnice, pretvori vhod v bite za shranjevanje in interpretira bite kot berljivo besedilo za prikaz na monitorju. Sistemska programska oprema se uporablja na številnih različnih vrstah naprav, kot so pametni televizorji, podporne naprave, virtualne komponente, komponente v oblaku in brezpilotna letala. Študenti lahko na primer raziskujejo napredovanje od napetosti do binarnega signala do logičnih vrat do seštevalnikov in tako naprej. Poznavanje posebnih, naprednih izrazov za računalniško arhitekturo, kot so BIOS, jedro ali vodilo, se na tej ravni ne pričakuje. | Računalniški sistemi | Strojna in programska oprema | Razvijanje in uporaba abstrakcij | 4.1 |
| 3A-CS-03 | 3A | 9-10 | **Razvijte smernice, ki posredujejo sistematične strategije za odpravljanje težav, ki jih lahko drugi uporabijo za prepoznavanje in odpravljanje napak.** | Odpravljanje zapletenih težav vključuje uporabo več virov pri raziskovanju, ocenjevanju in izvajanju potencialnih rešitev. Odpravljanje težav je odvisno tudi od izkušenj, na primer, ko ljudje prepoznajo, da je težava podobna tisti, ki so jo videli že prej, ali prilagodijo rešitve, ki so delovale v preteklosti. Primeri zapletenih strategij za odpravljanje težav vključujejo odpravljanje težav s povezljivostjo, prilagajanje konfiguracij in nastavitev sistema, zagotavljanje združljivosti strojne in programske opreme ter prenos podatkov iz ene naprave v drugo. Študenti bi lahko ustvarili diagram poteka, pomoč pri delu za zaposlenega v službi za pomoč uporabnikom ali strokovni sistem. | Računalniški sistemi | Odpravljanje težav | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.2 |
| 3A-NI-04 | 3A | 9-10 | **Ocenite razširljivost in zanesljivost omrežij z opisom razmerja med usmerjevalniki, stikali, strežniki, topologijo in naslavljanjem.** | Vsaki napravi je dodeljen naslov, ki jo enolično prepozna v omrežju. Usmerjevalniki delujejo tako, da primerjajo naslove IP, da določijo poti, ki jih morajo paketi udeležiti, da dosežejo svoj cilj. Stikala delujejo tako, da primerjajo naslove MAC in določijo, kateri računalniki ali segmenti omrežja bodo prejeli okvirje. Študenti bi lahko uporabili spletne omrežne simulatorje za eksperimentiranje s temi dejavniki. | Omrežja in internet | Omrežna komunikacija in organizacija | Razvijanje in uporaba abstrakcij | 4.1 |
| 3A-NI-05 | 3A | 9-10 | **Navedite primere za ponazoritev, kako lahko zlonamerna programska oprema in drugi napadi vplivajo na občutljive podatke.** | Varnost omrežja je odvisna od kombinacije strojne opreme, programske opreme in postopkov, ki nadzorujejo dostop do podatkov in sistemov. Potrebe uporabnikov in občutljivost podatkov določajo raven varnosti, ki se izvaja. Morebitne varnostne težave, kot so napadi zavrnitve storitve, izsiljevalska programska oprema, virusi, črvi, vohunska programska oprema in lažno predstavljanje, ogrožajo občutljive podatke. Študenti lahko razmišljajo o študijah primerov ali trenutnih dogodkih, v katerih so vlade ali organizacije zaradi tovrstnih napadov doživele uhajanje podatkov ali izgubo podatkov. | Omrežja in internet | Omrežna komunikacija in organizacija | Komuniciranje o računalništvu | 7.2 |
| 3A-NI-06 | 3A | 9-10 | **Priporočite varnostne ukrepe za obravnavo različnih scenarijev na podlagi dejavnikov, kot so učinkovitost, izvedljivost in etični vplivi.** | Varnostni ukrepi lahko vključujejo fizične varnostne žetone, dvofaktorsko preverjanje pristnosti in biometrično preverjanje. Morebitne varnostne težave, kot so napadi zavrnitve storitve, izsiljevalska programska oprema, virusi, črvi, vohunska programska oprema in lažno predstavljanje, ponazarjajo, zakaj je treba občutljive podatke varno shranjevati in prenašati. Pravočasen in zanesljiv dostop pooblaščenih uporabnikov do podatkovnih in informacijskih storitev, imenovan razpoložljivost, je zagotovljen z ustrezno pasovno širino, varnostnimi kopijami in drugimi ukrepi. Študenti bi morali sistematično oceniti izvedljivost uporabe računalniških orodij za reševanje danih problemov ali podproblemov, na primer z analizo stroškov in koristi. Sčasoma bi morali študenti v svoje ocene vključiti več dejavnikov, na primer, kako učinkovitost vpliva na izvedljivost ali ali predlagani pristop vzbuja etične pomisleke. | Omrežja in internet | Kibernetska varnost | Prepoznavanje in definiranje računalniških problemov | 3.3 |
| 3A-NI-07 | 3A | 9-10 | **Primerjajte različne varnostne ukrepe in upoštevajte kompromise med uporabnostjo in varnostjo računalniškega sistema.** | Varnostni ukrepi lahko vključujejo fizične varnostne žetone, dvofaktorsko preverjanje pristnosti in biometrično preverjanje, vendar izbira varnostnih ukrepov vključuje kompromise med uporabnostjo in varnostjo sistema. Potrebe uporabnikov in občutljivost podatkov določajo raven varnosti, ki se izvaja. Študenti lahko razpravljajo o politikah računalniške varnosti, ki veljajo na lokalni ravni, ki predstavljajo kompromis med uporabnostjo in varnostjo, kot je spletni filter, ki preprečuje dostop do številnih izobraževalnih spletnih mest, vendar ohranja omrežje kampusa varno. | Omrežja in internet | Omrežna komunikacija in organizacija | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.3 |
| 3A-NI-08 | 3A | 9-10 | **Pojasnite kompromise pri izbiri in izvajanju priporočil za kibernetsko varnost.** | Varnost omrežja je odvisna od kombinacije strojne opreme, programske opreme in postopkov, ki nadzorujejo dostop do podatkov in sistemov. Potrebe uporabnikov in občutljivost podatkov določajo raven varnosti, ki se izvaja. Vsak varnostni ukrep vključuje kompromise med dostopnostjo in varnostjo sistema. Študenti bi morali biti sposobni opisati, utemeljiti in dokumentirati izbire, ki jih sprejmejo, z uporabo terminologije, primerne za predvideno občinstvo in namen. Študenti bi lahko razpravljali o vprašanjih z vidika različnih občinstev, vključno s posamezniki, korporacijami, zagovorniki zasebnosti, varnostnimi strokovnjaki in vlado. | Omrežja in internet | Kibernetska varnost | Komuniciranje o računalništvu | 7.2 |
| 3A-DA-09 | 3A | 9-10 | **Prevajanje med različnimi predstavitvami bitov pojavov iz resničnega sveta, kot so znaki, številke in slike.** | Pretvorite na primer šestnajstiške barvne kode v decimalne odstotke, predstavitev ASCII/Unicode in logična vrata. | Podatki in analiza | Skladiščenje | Razvijanje in uporaba abstrakcij | 4.1 |
| 3A-DA-10 | 3A | 9-10 | **Ocenite kompromise pri organizaciji podatkovnih elementov in kje so podatki shranjeni.** | Ljudje se odločajo o tem, kako so podatkovni elementi organizirani in kje so shranjeni. Te izbire vplivajo na stroške, hitrost, zanesljivost, dostopnost, zasebnost in integriteto. Študenti morajo oceniti, ali je izbrana rešitev najprimernejša za določen problem. Študenti lahko razmislijo o stroških, hitrosti, zanesljivosti, dostopnosti, zasebnosti in celovitosti med shranjevanjem fotografskih podatkov na mobilni napravi in v oblaku. | Podatki in analiza | Skladiščenje | Prepoznavanje in definiranje računalniških problemov | 3.3 |
| 3A-DA-11 | 3A | 9-10 | **Ustvarite interaktivne vizualizacije podatkov s programskimi orodji, da bi drugim pomagali bolje razumeti pojave v resničnem svetu.** | Ljudje preoblikujejo, posplošujejo, poenostavljajo in predstavljajo velike nabore podatkov na različne načine, da vplivajo na to, kako drugi ljudje razlagajo in razumejo osnovne informacije. Primeri vključujejo vizualizacijo, združevanje, preureditev in uporabo matematičnih operacij. Ljudje uporabljajo programska orodja ali programiranje za ustvarjanje zmogljivih, interaktivnih vizualizacij podatkov in izvajanje vrste matematičnih operacij za preoblikovanje in analizo podatkov. Študenti bi morali modelirati pojave kot sisteme s pravili, ki urejajo interakcije v sistemu, in te modele oceniti glede na opazovanja v resničnem svetu. Na primer vedenje jate, čakalne vrste ali življenjski cikli. Tabele Google Fusion omogočajo dostop do ponazoritve podatkov v spletu. | Podatki in analiza | Vizualizacija in transformacija zbirke | Razvijanje in uporaba abstrakcij | 4.4 |
| 3A-DA-12 | 3A | 9-10 | **Ustvarite računalniške modele, ki predstavljajo razmerja med različnimi elementi podatkov, zbranih iz pojava ali procesa.** | Računalniški modeli napovedujejo procese ali pojave na podlagi izbranih podatkov in značilnosti. Količina, kakovost in raznolikost podatkov ter izbrane funkcije lahko vplivajo na kakovost modela in sposobnost razumevanja sistema. Napovedi ali sklepi se testirajo za preverjanje modelov. Študenti bi morali modelirati pojave kot sisteme s pravili, ki urejajo interakcije znotraj sistema. Študenti bi morali analizirati in oceniti te modele glede na opazovanja iz resničnega sveta. Študenti lahko na primer ustvarijo preprost model ekosistema proizvajalec-potrošnik z uporabo programskega orodja. Sčasoma bi lahko napredovali pri ustvarjanju bolj zapletenih in realističnih interakcij med vrstami, kot so plenilstvo, konkurenca ali simbioza, in ocenili model na podlagi podatkov, zbranih iz narave. | Podatki in analiza | Sklepanje in modeli | Razvijanje in uporaba abstrakcij | 4.4 |
| 3A-AP-13 | 3A | 9-10 | **Ustvarite prototipe, ki uporabljajo algoritme za reševanje računalniških problemov z izkoriščanjem predhodnega znanja študentov in osebnih interesov.** | Prototip je računalniški artefakt, ki prikazuje osnovno funkcionalnost izdelka ali procesa. Prototipi so uporabni za zgodnje pridobivanje povratnih informacij v procesu oblikovanja in lahko prinesejo vpogled v izvedljivost izdelka. Proces razvoja računalniških artefaktov zajema tako ustvarjalno izražanje kot raziskovanje idej za ustvarjanje prototipov in reševanje računalniških problemov. Študenti ustvarjajo artefakte, ki so osebno pomembni ali koristni za njihovo skupnost in širše. Študenti bi morali razviti artefakte kot odgovor na nalogo ali računalniški problem, ki kažejo zmogljivost, ponovno uporabo in enostavnost izvajanja algoritma. | Algoritmi in programiranje | Algoritmi | Ustvarjanje računalniških artefaktov | 5.2 |
| 3A-AP-14 | 3A | 9-10 | **Uporabite sezname za poenostavitev rešitev, posploševanje računskih problemov namesto ponavljajoče uporabe preprostih spremenljivk.** | Študenti bi morali biti sposobni prepoznati skupne značilnosti v več segmentih kode in nadomestiti en segment, ki uporablja sezname (matrike) za upoštevanje razlik. | Algoritmi in programiranje | Spremenljivke | Razvijanje in uporaba abstrakcij | 4.1 |
| 3A-AP-15 | 3A | 9-10 | **Utemeljite izbiro posebnih nadzornih struktur, kadar kompromisi vključujejo izvajanje, berljivost in uspešnost programa, ter pojasnite prednosti in slabosti sprejetih odločitev.** | Izvajanje vključuje izbiro programskega jezika, ki vpliva na čas in trud, potreben za izdelavo programa. Berljivost se nanaša na to, kako jasen je program drugim programerjem in ga je mogoče izboljšati z dokumentacijo. Razprava o uspešnosti je omejena na teoretično razumevanje časa izvedbe in zahtev po shranjevanju; kvantitativna analiza ni pričakovana. Nadzorne strukture na tej ravni lahko vključujejo pogojne izjave, zanke, upravljavce dogodkov in rekurzijo. Študenti lahko na primer primerjajo berljivost in uspešnost programa iterativnih in rekurzivnih izvedb postopkov, ki izračunavajo Fibonaccijevo zaporedje. | Algoritmi in programiranje | Kontrola | Prepoznavanje in definiranje računalniških problemov | 5.2 |
| 3A-AP-16 | 3A | 9-10 | **Oblikujte in iterativno razvijajte računalniške artefakte za praktičen namen, osebno izražanje ali za reševanje družbenega vprašanja z uporabo dogodkov za začetek navodil.** | V tem kontekstu ustrezni računalniški artefakti vključujejo programe, mobilne aplikacije ali spletne aplikacije. Dogodke lahko sproži uporabnik, na primer pritisk gumba, ali sistem, na primer sproženje časovnika. Na prejšnjih ravneh so se študenti naučili ustvarjati in klicati postopke. Tukaj učenci oblikujejo postopke, ki jih imenujejo dogodki. Učenci lahko ustvarijo mobilno aplikacijo, ki posodablja seznam bližnjih zanimivosti, ko naprava zazna, da je bila njena lokacija spremenjena. | Algoritmi in programiranje | Kontrola | Ustvarjanje računalniških artefaktov | 5.2 |
| 3A-AP-17 | 3A | 9-10 | **Razčlenite probleme na manjše komponente s sistematično analizo z uporabo konstruktov, kot so postopki, moduli in / ali objekti.** | Na tej ravni bi morali študenti razčleniti zapletene probleme na obvladljive podprobleme, ki bi jih lahko rešili s programi ali postopki, ki že obstajajo. Študenti bi lahko na primer ustvarili aplikacijo za reševanje problema skupnosti s povezovanjem s spletno bazo podatkov prek vmesnika za programiranje aplikacij (API). | Algoritmi in programiranje | Modularnost | Prepoznavanje in definiranje računalniških problemov | 3.2 |
| 3A-AP-18 | 3A | 9-10 | **Ustvarite artefakte z uporabo postopkov v programu, kombinacij podatkov in postopkov ali neodvisnih, vendar medsebojno povezanih programov.** | Računske artefakte lahko ustvarite z združevanjem in spreminjanjem obstoječih artefaktov ali z razvojem novih artefaktov. Primeri računalniških artefaktov vključujejo programe, simulacije, vizualizacije, digitalne animacije, robotske sisteme in aplikacije. Kompleksni programi so zasnovani kot sistemi medsebojno delujočih modulov, od katerih ima vsak določeno vlogo, ki se usklajuje za skupni splošni namen. Moduli omogočajo boljše upravljanje kompleksnih nalog. Na tej ravni je poudarek na razumevanju programa kot sistema z odnosi med moduli. Izbira implementacije, kot je programski jezik ali paradigma, se lahko razlikuje. Študenti bi lahko vključili knjižnice računalniškega vida, da bi povečali zmogljivosti robota ali izkoristili odprtokodne knjižnice JavaScript za razširitev funkcionalnosti spletne aplikacije. | Algoritmi in programiranje | Modularnost | Ustvarjanje računalniških artefaktov | 5.2 |
| 3A-AP-19 | 3A | 9-10 | **Sistematično oblikujte in razvijajte programe za širše občinstvo z vključitvijo povratnih informacij uporabnikov.** | Primeri programov lahko vključujejo igre, pripomočke in mobilne aplikacije. Študenti na nižjih ravneh zbirajo povratne informacije in pregledujejo programe. Na tej ravni bi morali študenti to storiti s sistematičnim postopkom, ki vključuje povratne informacije širokega občinstva. Študenti lahko ustvarijo anketo o zadovoljstvu uporabnikov in metode distribucije brainstorminga, ki bi lahko prinesle povratne informacije od raznolikega občinstva in dokumentirale postopek, ki so ga uporabili za vključitev izbranih povratnih informacij v revizije izdelkov. | Algoritmi in programiranje | Modularnost | Ustvarjanje računalniških artefaktov | 5.1 |
| 3A-AP-20 | 3A | 9-10 | **Ocenite licence, ki omejujejo ali omejujejo uporabo računalniških artefaktov pri uporabi virov, kot so knjižnice.** | Primeri licenc za programsko opremo vključujejo avtorske pravice, brezplačno programsko opremo in številne odprtokodne sheme licenciranja. Na prejšnjih ravneh so se študenti držali shem licenciranja. Na tej ravni bi morali razmisliti o posledicah licenciranja za svoje delo, zlasti pri vključevanju knjižnic in drugih virov. Študenti lahko razmislijo o dveh knjižnicah programske opreme, ki obravnavata podobne potrebe, in upravičijo svojo izbiro na podlagi knjižnice, ki ima najmanj omejevalno licenco. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.3 |
| 3A-AP-21 | 3A | 9-10 | **Ocenite in izboljšajte računalniške artefakte, da bodo bolj uporabni in dostopni.** | Preizkušanje in izpopolnjevanje je nameren in iterativen proces izboljšanja računalniškega artefakta. Ta postopek vključuje odpravljanje napak (prepoznavanje in odpravljanje napak) in primerjavo dejanskih rezultatov s predvidenimi rezultati. Študenti se morajo odzvati na spreminjajoče se potrebe in pričakovanja končnih uporabnikov ter izboljšati zmogljivost, zanesljivost, uporabnost in dostopnost artefaktov. Študenti bi lahko na primer vključili povratne informacije različnih končnih uporabnikov, da bi pomagali usmerjati velikost in postavitev menijev in gumbov v uporabniškem vmesniku. | Algoritmi in programiranje | Razvoj programa | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.3 |
| 3A-AP-22 | 3A | 9-10 | **Oblikujte in razvijajte računalniške artefakte, ki delajo v skupinskih vlogah z uporabo orodij za sodelovanje.** | Orodja za sodelovanje so lahko tako zapletena kot sistem za nadzor različic izvorne kode ali tako preprosta kot sodelovalni urejevalnik besedil. Vloge ekipe v programiranju v paru so voznik in navigator, vendar bi lahko bile bolj specializirane za večje ekipe. Ker programi postajajo vse bolj zapleteni, postaja izbira virov, ki pomagajo razvoju programa, vse pomembnejša in bi jo morali opraviti študenti. Učenci lahko delajo kot ekipa pri razvoju mobilne aplikacije, ki obravnava problem, ki je pomemben za šolo ali skupnost, pri čemer izberejo ustrezna orodja za vzpostavitev in upravljanje časovnice projekta; oblikovanje, skupna raba in revizija elementov grafičnega uporabniškega vmesnika; in spremljati načrtovane, tekoče in dokončane komponente. | Algoritmi in programiranje | Razvoj programa | Sodelovanje na področju računalništva | 2.4 |
| 3A-AP-23 | 3A | 9-10 | **Dokumentirajte odločitve o oblikovanju z uporabo besedila, grafike, predstavitev in / ali predstavitev pri razvoju kompleksnih programov.** | Kompleksni programi so zasnovani kot sistemi medsebojno delujočih modulov, od katerih ima vsak določeno vlogo, ki se usklajuje za skupni splošni namen. Ti moduli so lahko postopki v programu; kombinacije podatkov in postopkov; ali neodvisni, vendar medsebojno povezani programi. Razvoju kompleksnih programov pomagajo viri, kot so knjižnice in orodja za urejanje in upravljanje delov programa. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.2 |
| 3A-IC-24 | 3A | 9-10 | **Ocenite načine, kako računalništvo vpliva na osebne, etične, družbene, gospodarske in kulturne prakse.** | Računalništvo lahko izboljša, škoduje ali ohranja prakse. Primanjkljaji lastniškega kapitala, kot so minimalna izpostavljenost računalništvu, dostop do izobraževanja in možnosti usposabljanja, so povezani z večjimi sistemskimi problemi v družbi. Študenti bi morali biti sposobni oceniti dostopnost izdelka za široko skupino končnih uporabnikov, kot so ljudje, ki nimajo dostopa do širokopasovnih povezav ali imajo različne invalidnosti. Študenti bi morali začeti prepoznavati potencialno pristranskost med postopkom oblikovanja, da bi povečali dostopnost pri oblikovanju izdelkov. | Vplivi računalništva | Kultura | Spodbujanje vključujoče računalniške kulture | 1.2 |
| 3A-IC-25 | 3A | 9-10 | **Preizkusite in izboljšajte računalniške artefakte, da zmanjšate pristranskost in primanjkljaj lastniškega kapitala.** | Pristranskost lahko vključuje napačne predpostavke, ki so jih razvijalci naredili o svoji uporabniški bazi. Primanjkljaji lastniškega kapitala vključujejo minimalno izpostavljenost računalništvu, dostop do izobraževanja in priložnosti za usposabljanje. Študenti bi morali začeti prepoznavati morebitno pristranskost med postopkom oblikovanja, da bi povečali dostopnost pri oblikovanju izdelkov in se zavedali strokovno sprejetih standardov dostopnosti za ocenjevanje računalniških artefaktov za dostopnost. | Vplivi računalništva | Kultura | Spodbujanje vključujoče računalniške kulture | 1.2 |
| 3A-IC-26 | 3A | 9-10 | **Pokažite, kako se določen algoritem uporablja za probleme v različnih disciplinah.** | Računanje lahko deli funkcije z disciplinami, kot sta umetnost in glasba, z algoritemskim prevajanjem človeškega namena v artefakt. Študenti bi morali biti sposobni prepoznati resnične probleme, ki zajemajo več disciplin, kot je povečanje varnosti koles z novo tehnologijo čelade, in ki jih je mogoče rešiti računsko. | Vplivi računalništva | Kultura | Prepoznavanje in definiranje računalniških problemov | 3.1 |
| 3A-IC-27 | 3A | 9-10 | **Uporabite orodja in metode za sodelovanje na projektu, da povečate povezljivost ljudi v različnih kulturah in poklicnih področjih.** | Na številne vidike družbe, zlasti poklice, je vplivala stopnja komunikacije, ki jo omogoča računalništvo. Povečana povezanost med ljudmi v različnih kulturah in na različnih poklicnih področjih je spremenila naravo in vsebino številnih poklicev. Študenti bi morali raziskati različna orodja in metode sodelovanja, ki se uporabljajo za pridobivanje prispevkov članov ekipe, sošolcev in drugih, kot je sodelovanje na spletnih forumih ali lokalnih skupnostih. Študenti bi lahko na primer primerjali načine, kako bi lahko različna orodja družbenih medijev pomagala ekipi, da postane bolj kohezivna. | Vplivi računalništva | Družbene interakcije | Sodelovanje na področju računalništva | 2.4 |
| 3A-IC-28 | 3A | 9-10 | **Pojasnite koristne in škodljive učinke, ki jih lahko imajo zakoni o intelektualni lastnini na inovacije.** | Zakoni urejajo številne vidike računalništva, kot so zasebnost, podatki, lastnina, informacije in identiteta. Ti zakoni imajo lahko koristne in škodljive učinke, kot so pospeševanje ali zavlačevanje napredka v računalništvu in zaščita ali kršitev pravic ljudi. Mednarodne razlike v zakonodaji in etiki vplivajo na računalništvo. Na primer, zakoni, ki nalagajo blokiranje nekaterih spletnih mest za izmenjavo datotek, lahko zmanjšajo spletno piratstvo, vendar lahko omejijo pravico do dostopa do informacij. Požarni zidovi se lahko uporabljajo za blokiranje škodljivih virusov in zlonamerne programske opreme, lahko pa se uporabljajo tudi za cenzuro medijev. Študenti se morajo zavedati zakonov o intelektualni lastnini in biti sposobni razložiti, kako se uporabljajo za zaščito interesov inovatorjev in kako patentni troli zlorabljajo zakone za finančno korist. | Vplivi računalništva | Varnostno pravo in etika | Komuniciranje o računalništvu | 7.3 |
| 3A-IC-29 | 3A | 9-10 | **Pojasnite pomisleke glede zasebnosti, povezane z zbiranjem in ustvarjanjem podatkov z avtomatiziranimi postopki, ki uporabnikom morda niso očitni.** | Podatke je mogoče zbirati in združevati pri milijonih ljudi, tudi če ne sodelujejo aktivno z napravami za zbiranje podatkov ali v njihovi fizični bližini. Ta avtomatizirana in neočitna zbirka lahko povzroči pomisleke glede zasebnosti, kot so spletna mesta družbenih medijev, ki rudarijo račun, tudi če uporabnik ni na spletu. Drugi primeri vključujejo nadzorni video, ki se uporablja v trgovini za sledenje strankam za varnost ali informacije o nakupnih navadah ali spremljanje cestnega prometa za spreminjanje signalov v realnem času za izboljšanje učinkovitosti cest, ne da bi se vozniki zavedali. Metode in naprave za zbiranje podatkov se lahko razlikujejo glede na količino potrebnega pomnilnika, raven zbranih podrobnosti in hitrosti vzorčenja. | Vplivi računalništva | Varnostno pravo in etika | Komuniciranje o računalništvu | 7.2 |
| 3A-IC-30 | 3A | 9-10 | **Ocenite družbene in ekonomske posledice zasebnosti v kontekstu varnosti, zakona ali etike.** | Zakoni urejajo številne vidike računalništva, kot so zasebnost, podatki, lastnina, informacije in identiteta. Mednarodne razlike v zakonodaji in etiki vplivajo na računalništvo. Študenti lahko pregledajo študije primerov ali trenutne dogodke, ki predstavljajo etično dilemo, ko je posameznikova pravica do zasebnosti v nasprotju z varnostjo, zaščito ali blaginjo skupnosti. | Vplivi računalništva | Varnostno pravo in etika | Komuniciranje o računalništvu | 7.3 |
| 3B-CS-01 | 3B | 11-12 | **Kategorizirajte vloge programske opreme operacijskega sistema.** | Primeri vlog lahko vključujejo upravljanje pomnilnika, shranjevanje / pridobivanje podatkov, upravljanje procesov in nadzor dostopa. | Računalniški sistemi | Strojna in programska oprema | Komuniciranje o računalništvu | 7.2 |
| 3B-CS-02 | 3B | 11-12 | **Ponazorite načine, kako računalniški sistemi izvajajo logiko, vhod in izhod prek komponent strojne opreme.** | Primeri komponent lahko vključujejo logična vrata in IO pine. | Računalniški sistemi | Odpravljanje težav | Komuniciranje o računalništvu | 7.2 |
| 3B-NI-03 | 3B | 11-12 | **Opišite težave, ki vplivajo na delovanje omrežja (npr. pasovna širina, obremenitev, zamuda, topologija).** | Priporočite uporabo brezplačnih spletnih simulatorjev omrežja, da raziščete, kako te težave vplivajo na delovanje omrežja. | Omrežja in internet | Omrežna komunikacija in organizacija | Komuniciranje o računalništvu | 7.2 |
| 3B-NI-04 | 3B | 11-12 | **Primerjajte načine, kako razvijalci programske opreme ščitijo naprave in podatke pred nepooblaščenim dostopom.** | Primeri varnostnih pomislekov, ki jih je treba upoštevati: strategije šifriranja in preverjanja pristnosti, varno kodiranje in varovanje ključev. | Omrežja in internet | Kibernetska varnost | Komuniciranje o računalništvu | 7.2 |
| 3B-DA-05 | 3B | 11-12 | **Uporabite orodja in tehnike za analizo podatkov za prepoznavanje vzorcev v podatkih, ki predstavljajo kompleksne sisteme.** | Prepoznajte na primer trende v naboru podatkov, ki predstavljajo interakcije v družabnih medijih, ocene filmov ali nakupovalne vzorce. | Podatki in analiza | Vizualizacija in transformacija zbirke | Razvijanje in uporaba abstrakcij | 4.1 |
| 3B-DA-06 | 3B | 11-12 | **Izberite orodja in tehnike zbiranja podatkov za ustvarjanje podatkovnih nizov, ki podpirajo zahtevek ali sporočajo informacije.** |  | Podatki in analiza | Vizualizacija in transformacija zbirke | Komuniciranje o računalništvu | 7.2 |
| 3B-DA-07 | 3B | 11-12 | **Ocenite sposobnost modelov in simulacij za preizkušanje in podporo izpopolnjevanju hipotez.** |  | Podatki in analiza | Sklepanje in modeli | Razvijanje in uporaba abstrakcij | 4.4 |
| 3B-AP-08 | 3B | 11-12 | **Opišite, kako umetna inteligenca poganja veliko programske opreme in fizičnih sistemov.** | Primeri vključujejo prikazovanje digitalnih oglasov, samovozeče avtomobile in zaznavanje goljufij s kreditnimi karticami. | Algoritmi in programiranje | Algoritmi | Komuniciranje o računalništvu | 7.2 |
| 3B-AP-09 | 3B | 11-12 | **Implementirajte algoritem umetne inteligence za igranje igre proti človeškemu nasprotniku ali rešitev problema.** | Igre niso nujno zapletene. Zadostujejo preproste igre ugibanja, Tic-Tac-Toe ali preprosti robotski ukazi. | Algoritmi in programiranje | Algoritmi | Ustvarjanje računalniških artefaktov | 5.3 |
| 3B-AP-10 | 3B | 11-12 | **Uporabite in prilagodite klasične algoritme za reševanje računskih problemov.** | Primeri lahko vključujejo razvrščanje in iskanje. | Algoritmi in programiranje | Algoritmi | Razvijanje in uporaba abstrakcij | 4.2 |
| 3B-AP-11 | 3B | 11-12 | **Ocenite algoritme glede na njihovo učinkovitost, pravilnost in jasnost.** | Primeri lahko vključujejo razvrščanje in iskanje. | Algoritmi in programiranje | Algoritmi | Razvijanje in uporaba abstrakcij | 4.2 |
| 3B-AP-12 | 3B | 11-12 | **Primerjajte in primerjajte temeljne podatkovne strukture in njihovo uporabo.** | Primeri lahko vključujejo nize, sezname, matrike, sklade in čakalne vrste. | Algoritmi in programiranje | Spremenljivke | Razvijanje in uporaba abstrakcij | 4.2 |
| 3B-AP-13 | 3B | 11-12 | **Ponazorite potek izvajanja rekurzivnega algoritma.** |  | Algoritmi in programiranje | Kontrola | Prepoznavanje in definiranje računalniških problemov | 3.2 |
| 3B-AP-14 | 3B | 11-12 | **Konstruirajte rešitve za probleme z uporabo komponent, ki jih ustvarijo študenti, kot so postopki, moduli in / ali predmeti.** | Objektno usmerjeno programiranje je na tej ravni neobvezno. Težave je mogoče dodeliti ali izbrati učenec. | Algoritmi in programiranje | Modularnost | Ustvarjanje računalniških artefaktov | 5.2 |
| 3B-AP-15 | 3B | 11-12 | **Analizirajte obsežen računski problem in določite posplošljive vzorce, ki jih je mogoče uporabiti za rešitev.** | Ko se učenci srečujejo s kompleksnimi problemi iz resničnega sveta, ki zajemajo več disciplin ali družbenih sistemov, bi morali kompleksne probleme razgraditi na obvladljive podprobleme, ki bi jih lahko rešili s programi ali postopki, ki že obstajajo. Študenti bi lahko na primer ustvarili aplikacijo za reševanje problema skupnosti s povezovanjem s spletno bazo podatkov prek vmesnika za programiranje aplikacij (API). | Algoritmi in programiranje | Modularnost | Razvijanje in uporaba abstrakcij | 4.1 |
| 3B-AP-16 | 3B | 11-12 | **Pokažite ponovno uporabo kode z ustvarjanjem programskih rešitev z uporabo knjižnic in API-jev.** | Knjižnice in API-ji so lahko na primer študentske ali običajne grafične knjižnice ali API-ji zemljevidov. | Algoritmi in programiranje | Modularnost | Ustvarjanje računalniških artefaktov | 5.3 |
| 3B-AP-17 | 3B | 11-12 | **Načrtujte in razvijajte programe za širše občinstvo z uporabo procesa življenjskega cikla programske opreme.** | Procesi lahko vključujejo agilne, spiralne ali slapove. | Algoritmi in programiranje | Razvoj programa | Ustvarjanje računalniških artefaktov | 5.1 |
| 3B-AP-18 | 3B | 11-12 | **Pojasnite varnostne težave, ki lahko privedejo do ogroženih računalniških programov.** | Pogoste težave vključujejo na primer pomanjkanje preverjanja meja, slabo preverjanje vnosa in krožne reference. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.2 |
| 3B-AP-19 | 3B | 11-12 | **Razvijajte programe za več računalniških platform.** | Primeri platform lahko vključujejo: namizni računalnik, splet ali mobilno napravo. | Algoritmi in programiranje | Razvoj programa | Ustvarjanje računalniških artefaktov | 5.2 |
| 3B-AP-20 | 3B | 11-12 | **Uporabite sisteme za nadzor različic, integrirana razvojna okolja (IDE) ter orodja in prakse za sodelovanje (dokumentacija kode) v skupinskem projektu programske opreme.** | Skupinske projekte programske opreme je mogoče dodeliti ali izbrati študent. | Algoritmi in programiranje | Razvoj programa | Sodelovanje na področju računalništva | 2.4 |
| 3B-AP-21 | 3B | 11-12 | **Razviti in uporabiti vrsto testnih primerov, da preverite, ali program deluje v skladu s svojimi specifikacijami zasnove.** | Na tej ravni se od študentov pričakuje, da izberejo svoje testne primere. | Algoritmi in programiranje | Razvoj programa | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.1 |
| 3B-AP-22 | 3B | 11-12 | **Spremenite obstoječi program, da dodate dodatne funkcije in razpravljate o namernih in nenamernih posledicah (npr. prekinitev drugih funkcij).** | Na primer, spremembe v podpisu metode ali funkcije lahko prekinejo klicanje te metode drugje v sistemu. | Algoritmi in programiranje | Razvoj programa | Ustvarjanje računalniških artefaktov | 5.3 |
| 3B-AP-23 | 3B | 11-12 | **Ocenite ključne lastnosti programa s postopkom, kot je pregled kode.** | Primeri lastnosti lahko vključujejo pravilnost, uporabnost, berljivost, učinkovitost, prenosljivost in razširljivost. | Algoritmi in programiranje | Razvoj programa | Preizkušanje in izpopolnjevanje računalniških artefaktov | 6.3 |
| 3B-AP-24 | 3B | 11-12 | **Primerjajte več programskih jezikov in razpravljajte o tem, kako so njihove lastnosti primerne za reševanje različnih vrst problemov.** | Primeri funkcij vključujejo bloke v primerjavi z besedilom, zamik v primerjavi s zavitim oklepajem in visoko raven v primerjavi z nizko ravnjo. | Algoritmi in programiranje | Razvoj programa | Komuniciranje o računalništvu | 7.2 |
| 3B-IC-25 | 3B | 11-12 | **Ocenite računalniške artefakte, da povečate njihove koristne učinke in zmanjšate škodljive učinke na družbo.** |  | Vplivi računalništva | Kultura | Preizkušanje in izpopolnjevanje računalniških artefaktov, spodbujanje vključujoče računalniške kulture | 6.1, 1.2 |
| 3B-IC-26 | 3B | 11-12 | **Ocenite vpliv pravičnosti, dostopa in vpliva na porazdelitev računalniških virov v globalni družbi.** |  | Vplivi računalništva | Kultura | Spodbujanje vključujoče računalniške kulture | 1.2 |
| 3B-IC-27 | 3B | 11-12 | **Napovedati, kako se lahko razvijejo računalniške inovacije, ki so revolucionirale vidike naše kulture.** | Področja, ki jih je treba upoštevati, lahko vključujejo izobraževanje, zdravstveno varstvo, umetnost / zabavo in energijo. | Vplivi računalništva | Kultura | Komuniciranje o računalništvu | 7.2 |
| 3B-IC-28 | 3B | 11-12 | **Razpravljajte o zakonih in predpisih, ki vplivajo na razvoj in uporabo programske opreme.** |  | Vplivi računalništva | Varnostno pravo in etika | Prepoznavanje in definiranje računalniških problemov, komuniciranje o računalništvu | 3.3, 7.3 |