

KATALOG ZNANJA

1. IME PREDMETA

PODATKOVNE STRUKTURE IN ALGORITMI

2. SPLOŠNI CILJI

Splošni cilji predmeta so:

- usposobiti za implementacijo podatkovnih struktur in algoritmov za reševanje konkretnih problemov;
- predstaviti namen, kompleksnost ter rabo podatkovnih struktur ter algoritmov za delo v programskem inženirstvu;
- usposobiti za razreševanje težav, kritično mišljenje, analitično razmišljanje, sprejemanje odločitev, učinkovito komunikacijo;
- razviti digitalno in računsko pismenost ter kreativnost;
- naučiti načrtovanja in določanja prioritet, samoupravljanja, samostojnega učenja ter samorefleksije;
- usposobiti za projektno vodenje.

Specifično strokovno usmerjeni cilji so:

- načrtovati in implementirati podatkovne strukture z uporabo algoritmov;
- obvladati specifikacijo abstraktnih podatkovnih tipov in konkretnih podatkovnih struktur;
- izvajati postopke formalne notacije, ovijanja oz. inkapsulacije podatkov, implementacije programskih jezikov, asimptotskega testiranja ter izvajanja primerjalnih preizkusov oz. benchmarking;
- določiti zahteve programskih podatkov, abstraktnih podatkovnih tipov;
- uporabiti paradigme programiranja za izdelavo različnih algoritmov sortiranja, iskati, navigirati in vrednotiti učinkovitost izdelanih algoritmov.

3. PREDMETNO SPECIFIČNE KOMPETENCE

Pri predmetu študenti poleg generičnih pridobijo naslednje kompetence:

1. analiziranje abstraktnih podatkovnih tipov, konkretnih podatkovnih struktur ter algoritmov;
2. določanje abstraktnih podatkovnih tipov ter algoritmov v formalnem zapisu;
3. implementiranje kompleksnih algoritmov ter podatkovnih struktur;
4. vrednotenje učinkovitosti algoritmov in podatkovnih struktur.

4. OPERATIVNI CILJI

INFORMATIVNI CILJI	FORMATIVNI CILJI
1. Analiziranje abstraktnih podatkovnih tipov, konkretnih podatkovnih struktur ter algoritmov	
<ul style="list-style-type: none"> • definira abstraktne podatkovne tipe (ADT); • analizira način formalne notacije abstraktnih podatkovnih tipov; • razčleni pojme podatkovnih struktur (nizi, množice, linearni sezname, čakalne vrste, drevesa ter tipi, npr. aktivni, pasivni, rekurzivni, i. d.); • definira delovanje pomnilnika in proces implementacije ukazov v sklic funkcij računalnika; • analizira vrste algoritmov ter njihov namen (npr. rekurzivni, dinamični, naključni, metode: deli in vladaj, razveji in imeji – branch and bound, požrešna metoda, napad z grobo silo, i. d.); • ovrednoti funkcije algoritmov ter njihovo uporabo v programskem inženirstvu (npr. sortiranje, vrivanje, združevanje, kopičenje, linearno ter binarno iskanje, definira pojem iskalnega drevesa, i. d.); • oceni delovanje operacije iskanja drevesa najkrajših poti dveh algoritmov. 	<ul style="list-style-type: none"> • izdelava načrt za specifikacijo podatkovnih struktur ter predstavi ustrezne operacije, ki se lahko izvedejo znotraj strukture; • izdelava poročila primerov delovanja konkretnih podatkovnih struktur za čakalno vrsto FIFO; • izdelava predstavitev o učinkovitosti delovanja dveh algoritmov za urejanje podatkov;
2. Določanje abstraktnih podatkovnih tipov ter algoritmov v formalnem zapisu	
<ul style="list-style-type: none"> • definira postopek načrtovanja ter določanja abstraktnih podatkovnih tipov s formalno notacijo (npr. ASN.1); • opiše uporabo imperativne definicije za določanje abstraktnih podatkovnih tipov za določen sklad programske opreme; • analizira uporabo programskega jezika za neizvršljive datoteke (npr. SDL, VDM, i. d.); • razčleni kompleksnost izdelovanja programske opreme (npr. vzorci načrtovanja, vzporednost, uporabniški vmesniki, inkapsulacija, skrivanje informacij, učinkovitost, i. d.); 	<ul style="list-style-type: none"> • izdelava predstavitev o prednostih inkapsulacije ter skrivanja podatkov pri uporabi abstraktnih podatkovnih tipov; • izdelava ter implementira predpogoje ter pogoje delovanja programske opreme ter pogoje za določevanja napak pri delovanju.

<ul style="list-style-type: none"> • ovrednoti ter obrazloži pomen imperativnih abstraktnih podatkovnih tipov kot osnovo za orientacijo objektov. 	
<p>3. Implementiranje kompleksnih algoritmov ter podatkovnih struktur</p>	
<ul style="list-style-type: none"> • definira postopek implementacije podatkovnih struktur (pojmi večdimenzionalne polje, povezani sezname, skladi, vrste, drevesa, zgoščevalne tabele, kopice, grafi) ter način reševanja podrobno definiranih problemov; • razčleni vrste algoritmov ter način implementacije algoritem v programsko inženirstvo (npr. funkcije sortiranja, iskanja, iskanja po drevesu, iskanje po seznamu, kopičenja, spreminjanja nizov, načrtovalni in rekurzivni algoritmi, i. d.); • analizira prednosti inkapsulacije ter skrivanja podatkov pri uporabi abstraktnih podatkovnih tipov; • ovrednoti rabo izvršljivega programskega jezika za implementacijo algoritmov ter podatkovnih baz v delo programskega inženirja. 	<ul style="list-style-type: none"> • vstavi napreden abstrakten podatkovni tip ter algoritem za izvršni programski jezik, da reši podrobno definiran problem; • izdelava navodila za ravnanje z napakam ter poročila o rezultatih dela; • izdelava predstavitev o vrednotenju kompleksnosti implementiranega abstraktnega podatkovnega tipa ter algoritma.
<p>4. Vrednotenje učinkovitosti algoritmov in podatkovnih struktur</p>	
<ul style="list-style-type: none"> • utemelji uporabo knjižnic podatkovnih struktur (DSL) – omejitve, ročni izbor podatkovnih struktur, vrednost N, O notacija; • razčleni postopke merjenja učinkovitosti algoritmov (primerjava delovanja, odvisnosti med prevajalnikom in tolmačem, stopnje vzporednosti, pregled porabe sredstev); • definira rabo asimptotske analize za ocenjevanje učinkovitosti algoritma; • analizira prednosti implementacije neodvisnih podatkovnih struktur. • ovrednoti prednosti in slabosti specifikacije abstraktnih podatkovnih tipov. 	<ul style="list-style-type: none"> • izdelava dva praktična postopka merjenja učinkovitosti algoritma. • predstavi najmanj dva postopka merjenja učinkovitosti delovanja algoritmov, z uporabo praktičnih primerov implementacije.

5. OBVEZNOSTI ŠTUDENTOV IN POSEBNOSTI V IZVEDBI

Število kontaktnih ur: 80 ur (40 ur predavanj, 40 ur seminarskih vaj).

Število ur samostojnega dela študenta: 180 (študij literature, delo s programsko opremo, delo z besedilom, študij primerov, priprava pisnih besedil, predstavitev ...).