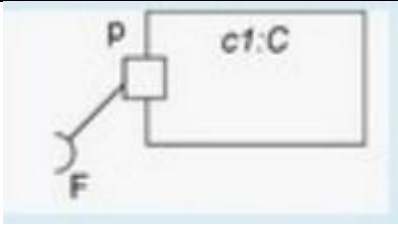



PITANJE	ODOGOVOR
Za tehnike simulacije i ispitivanja programa u procesu dokazivanja ispravnosti vrijedi:	Može se dokazati postojanje, ali ne i nepostojanje pogreške
Koje su generičke aktivnosti u svakom procesu inženjerstva zahtjeva?	Specifikacija Razvoj i oblikovanje Validacija i verifikacija Evolucija
U iteracije spiralnog modela procesa inženjerstva zahtjeva ne pripadaju:	Izrada prototipa Studija izvedivosti
U načine dokumentiranja zahtjeva sustava koji se koriste u praksi NE spada:	UML dijagram razreda
 <p>Slika</p>	Objekt c1 je tipa komponenta, port p zahtjeva svojstva definirana sučeljem F
bla grupiranja (engl. grouping things) odgovara kao UML opis:	Paketi
U sekvencijskom dijagramu UML kosa ili sl. padajuća linija poruke predstavlja:	Poruku koja ima vremensko trajanje
Kod oblikovanja kojih zahtjeva su bla UML obrasci uporabe:	funkcionalnih zahtjeva
Odnos između A i B (<< extend >>)	Obrazac A proširuje osnovni način uporabe obrasca B
	
Što je potrebno uključiti u dokument oblikovanja arhitekture programske potpore?	Prioritete koji su vodili proces oblikovanja Referenciju prema dokumentu zahtjeva Opis s najviše razine promatranja
Pristup zasnovan na razvoju oblikovanju i isporuci vrlo malih inkremenata funkcionalnosti svojstven je kojoj metodologiji razvoja programske potpore?	Agilnoj
Koji je model procesa programskog inženjerstva najpogodniji za male i srednje interaktivne sustave ?	Evolucijski
Koje od navedenih tvrdnji vrijede za spiralni model razvoja programske podrške ?	Sektori u spirali slijede vodopadni model Smanjuje se rizik razvoja Brza izrada prototipa
U drugu fazu unificiranog procesa (Faza elaboracije) ne spada:	Oblikovanje produkta Definiranje dosega projekta
Koja od navedenih je temeljna značajka vodopadnog modela?	Pojedina faza se mora dovršiti prije pokretanja nove faze
Što znači „oblikuj po ugovoru“	Sve metode imaju ugovor s pozivateljem

Koja je temeljna značajka svojstvena modernom načinu oblikovanja PP-e (PP = programska potpora nadalje)	Uvođenje modela sustava
Princip oblikovanja kojemu jecilj osigurati sakrivanje ili odgodu razmatranja detalja te time i smanjenje složenosti naziva se	Zadrži razinu apstrakcije
Razred može implementirati više sučelja:	Točno
Kod preopterećivanja (overloading) metode imaju _____ naziv , a _____ broj i tip parametara	istog , različitog
Metode razreda deklariraju se unutar razreda ključno riječi _____ i pozivaju se preko _____	static , imena razreda
Moguće je kreirati instance apstraktnog razreda ali je tada potreban modifikator abstract:	Netočno
Kako u objektno usmjerenom terminologiji nazivamo predložak (specifikaciju) koja definira vrstu objekta ?	Razred (Class)
Konstrukture upotrebljavamo za:	Inicijalizaciju novih objekata
_____ varijable određuje koje objekte ta varijabla može referencirati	tip ? PLS točan odg na FER2 postajte ako ovo nije
U procesu oblikovanja PP arhitektura programa rezultat je _____ - level , a implementacija _____ - level dizajna	visoki, niski PLS točan odgovor na FER2 postajte ako ovo nije
Koje su dvije osobine zajedničke svim objektima u OO?	Stanje i ponašanje
Koje od naveddenih tvrdnji vrijede za odnos agregacije u UML dijagramu:	agregat predstavlja IS-PART-OF odnos, agregacija je slabija od kompozicije
Osnovu opisa funkcionalnih zahtjeva u dokumentaciji projekta čine:	UML dijagrami obrazaca uporabe UML sekvencijski dijagrami
Naredba Subversion sustava kojom se promjene lokalnog repozitorija prenose na središnji repozitorij: 90 % sam siguran da bi ovo moglo doći – novo pitanje	checkin
Koje su temeljne značajke dobrog programskog produkta ?	Mora se moćo idržavati Mora biti prihvatljiv korisniku Mora biti pouzdan
Zahtjevi sustava mogu se izraziti:	Sve od navedenog (Strukturirnim..., Specijalnim..., Matematičkom..., Grafičkom...)
U iteracije spiralnog modela procesa inženjerstva zahtjeva ne pripadaju:	Studija izvedivosti Izrada protipa
Etnografija je:	zahtjevi izvedeni temeljem istraživanja kako ljudi stvarno rade
Osnovni elementi UML-a su:	stvari, relacije, objekti
Strukturne stvari u jeziku UML služe za opis:	razredi, sučelja, slučajevi korištenja, komponentni čvorovi, aktori i sl. Lakše vam je ovo zapamtiti kao najduži pdgovor haha

Za opis upravljačkog toka UML sekvencijskim dijagramom vrijede tvrdnje:	Moguće je opisati uvjetno grananje i petlje
Kod oblikovanja kojih zahtjeva se koriste UML obrasci uporabe?	Funkcionalnih zahtjeva
Koji od navedenih tipa poveznica NE postoji u dijagramu obrazaca uporabe ?	Kompozicija (composition)
Princip višestrukog korištenja u oblikovanju PP koristi se u :	SVE ponuđeno je točno ponovnom korištenju fragmenata rješenja, konzistencije, dijelova arhitekture, arhitekture, cjelokupne arhitekture sustava
Koje se tri aktivnosti ponavljaju tijekom razvoja u evolucijskom modelu?	Specifikacija, razvoj, validacija
Što nije sektor u spiralnom modelu iteracija PI-a?	Pridjeljivanje zahtjeva inkrementima
Koja je temeljna značajka vodopadnog modela?	Pojedina faza se mora dovršiti prije pokretanja nove faze
Obrasci uporabe specificiraju _____, a arhitektura _____ izvedbu	PLS ODGOVOR NA FER2 postajte, ne znam ovo
Princip oblikovanja koji se temelji na obradi svih slučajeva u kojima se neka komponenta može neprikladno upotrijebiti naziva se:	Konzervativno oblikovanje
Princip oblikovanja povećanja ponovne uporabivosti u pristupu oblikovanju je komplementaran principu	Povećaj uporabu postojećeg
Koji od navedenih pojmova NE predstavlja neki od osnovnih principa objektne paradigme	Nasljeđivanje (možda je ona napredan haha)
Svojstvo da različiti razredi imaju funkcije istog imena nazivamo:	Polimorfizam
sad je tu slika koda koji je nebitan ionako	
Što se događa tijekom izvođenja OO aplikacije?	Dio modela objekata se nalazi u memoriji i dinamički se stvaraju/brišu
Proces kojim jedan objekt može steći svojstva nekog drugog objekta nazivamo	Nasljeđivanje
Odnos kompozicije i agregacije u UML dijagramu razreda:	Kompozicija je jača od agregacije
UML dijagrami koji nisu nužni u inačici 1.0 su	Dijagram komponenti Dijagram Aktivnosti Dijagram razmještaja
Neki od alata za upravljanje inačicama datoteka unutar projekta su:	GIT, TFS(Team Foundation Server)
Kako se zove test u OOP kod provjere pravilnog nasljeđivanja	IS-A

MOJA PITANJA (vrlo jerojatna pogotovo iz SVN-a)	ODGOVORI
Što je Apache Subversion	Sustav za kontrolu inačica PP pod Apache licencom
Kako se zove test u OOP kod provjere pravilnog nasljeđivanja	„IS-A“
Koja operacija mijenja lokalnu radnu kopiju kako bi bila identična stanju središnjeg repozitorija	Update
Koja operacija promjene nastale u lokalnoj radnoj kopiji izvozi u središnji repozitorij	Commit
Prilikom stvaranja lokalne radne kopije (prvo dohvaćanje sadržaja središnjeg repozitorija) koristi se operacija:	Checkout
Koja operacija rješava sukob odbacivanjem vlastitih promjena	Revert
Operacija kojom se odabire koja će se inačica zadržati kod sukoba	Resolve
Apache SVN podržava isključivo:	komandno-linijski način rada
Tortoise SVN ima integrirano:	grafičko sučelje
Model u kojem ne postoji jedinstveni središnji repozitorij s kojim se klijenti sinkroniziraju, već svaka radna kopija repozitorija predstavlja repozitorij za sebe	Raspodijeljeni model
GIT podrazumijeva rad s dva repozitorija:	lokalnim i udaljenim