

Образац за критеријуме оцењивања по наставним темама

Назив наставног предмета: **Објектно оријентисано програмирање**

Разред: III Образовни профил: Ученици са посебним способностима за рачунарство и информатику

<p>ЕЛЕМЕНТИ ОЦЕЊИВАЊА</p>	<p>Назив наставне теме/модула/области¹:</p> <p>ОСНОВНИ ПОЈМОВИ ОБЈЕКТНО ОРИЈЕНТИСАНОГ ПРОГРАМИРАЊА</p>
<p>Очекивани исходи који ће се оцењивати (општи)</p>	<p>Ученик ће бити у стању да:</p> <ul style="list-style-type: none"> – наброји основне карактеристике објектно оријентисане парадигме; – употреби готове класе и објекте у креирању апликација; – наведе разлику између класе и објекта; – објасни поступак моделовања на конкретним примерима; – опише интерфејс задате класе; – демонстрира концепт енкапсулације и објасни права приступа елементима класе; – напише класу са потребним атрибутима и методама; – напише конструкторе и деструктор у класи; – осмисли и имплементира решење задатка коришћењем новедефинисане класе и њених објеката; – осмисли и имплементира класу коју затим користи у више различитих апликација; – за задати проблем креира једноставан систем повезаних класа и апликацију којом се тај проблем решава;
<p>КРИТЕРИЈУМ ОЦЕЊИВАЊА</p> <p>– у складу са операционализованим исходима</p>	
<p>оцена (5)</p>	<ul style="list-style-type: none"> • Основни појмови објектно оријентисаног програмирања: Ученик одлично разуме основне концепте ООП-а, као што су класе, објекти, енкапсулација, апстракција, и у потпуности је способан да објасни и примени ове појмове у разним ситуацијама. Самостално развија сложеније програме користећи ове концепте, уз поштовање најбољих пракси кодирања.
<p>оцена (4)</p>	<ul style="list-style-type: none"> • Основни појмови објектно оријентисаног програмирања: Ученик разуме основне ООП концепте и може их применити у већини случајева, али повремено прави грешке у сложенијим ситуацијама. Програми које развија углавном функционишу исправно, али могу имати мање недостатке или бити мање оптимизовани.
<p>оцена (3)</p>	<ul style="list-style-type: none"> • Основни појмови објектно оријентисаног програмирања: Ученик има основно разумевање ООП концепата, али повремено греши у примени. Програми које развија функционишу, али уз више недостатака или уз значајне проблеме у оптимизацији и коду.
<p>оцена (2)</p>	<ul style="list-style-type: none"> • Основни појмови објектно оријентисаног програмирања: Ученик има основно познавање ООП-а, али показује озбиљне недостатке у разумевању и примени концепата. Програми често садрже озбиљне грешке и не функционишу исправно без значајних интервенција.

оцена (1)	<ul style="list-style-type: none"> Основни појмови објектно оријентисаног програмирања: Ученик нема адекватно разумевање основних ООП концепата и није способан да их примени у пракси. Програми које развија су нефункционални, са бројним озбиљним грешкама.
Напомена	

ЕЛЕМЕНТИ ОЦЕЊИВАЊА	<p>Назив наставне теме/модула/области¹:</p> <p>ПРИНЦИПИ НАСЛЕЂИВАЊА И ПОЛИМОРФИЗМА</p>
Очекивани исходи који ће се оцењивати (општи)	<p>Ученик ће бити у стању да:</p> <ul style="list-style-type: none"> – опише концепт наслеђивања и однос „врста-од“; – наброји примере неких наткласа и њихових изведених класа; – на примерима објасни права приступа елементима основне класе из објекта изведене класе; – дефинише конструкторе и деструкторе у наткласи и изведеним класама; – објасни принцип полиморфизма; – напише виртуалне методе у оквиру дефиниција класа; – дефинише апстрактне методе и апстрактне класе; – на примерима илуструје разлику између апстрактне класе и интерфејса;
<p>КРИТЕРИЈУМ ОЦЕЊИВАЊА</p> <p>– у складу са операционализованим исходима</p>	
оцена (5)	<ul style="list-style-type: none"> Принципи наслеђивања и полиморфизма: Ученик детаљно разуме и може применити принципе наслеђивања, укључујући вишеструко и хијерархијско наслеђивање. Полиморфизам је јасно схваћен и правилно примењен, ученик користи динамичку и статичку полиморфију где је потребно. Способан је да реши проблеме користећи сложене структуре наслеђивања и полиморфизма.
оцена (4)	<ul style="list-style-type: none"> Принципи наслеђивања и полиморфизма: Ученик разуме и примењује принципе наслеђивања и полиморфизма, али можда не увек на најфикаснији начин. Постоји добро разумевање, али са повременим грешкама у имплементацији комплекснијих структура.
оцена (3)	<ul style="list-style-type: none"> Принципи наслеђивања и полиморфизма: Ученик је упознат са принципима наслеђивања и полиморфизма, али често прави грешке у примени. Постоје проблеми у имплементацији комплекснијих структура, са честим неспоразумима у вези са полиморфизмом.
оцена (2)	<ul style="list-style-type: none"> Принципи наслеђивања и полиморфизма: Ученик слабо разуме принципе наслеђивања и полиморфизма, и често греша приликом њихове имплементације. Проблеми у имплементацији су чести и значајно утичу на функционалност програма.
оцена (1)	<ul style="list-style-type: none"> Принципи наслеђивања и полиморфизма: Ученик не разуме принципе наслеђивања и полиморфизма и није способан да их примени. Грешке су константне и онемогућавају правилну функционалност програма.
Напомена	

<p>ЕЛЕМЕНТИ ОЦЕЊИВАЊА</p>	<p>Назив наставне теме/модула/области¹:</p> <p>ПРОЈЕКТНИ ЗАДАТАК</p>
<p>Очекивани исходи који ће се оцењивати (општи)</p>	<p>Ученик ће бити у стању да:</p> <ul style="list-style-type: none"> – осмисли и имплементира решење задатка коришћењем једне класе и класа изведених из ње; – за дати проблем уочи основне објекте и везе између њих, развије и имплементира хијерархије класа и интерфејса, помоћу којих могу да се реше тај и њему сродни проблеми; – тимски или индивидуално, а уз помоћ наставника, дефинише сложенији проблем за чије решавање осмишља и користи хијерархије класа; – тимски или индивидуално развије и приказује идејно решење проблема; – тимски или индивидуално развије план рада и начин праћења успешности реализације плана; – развије решење изабраног проблема или дела за који је задужен; – пише документацију; – креира презентацију и презентује решење пројектног рада; – вреднује своју улогу при изради пројектног задатка и активности за које је био задужен.
<p>КРИТЕРИЈУМ ОЦЕЊИВАЊА</p> <p>– у складу са операционализованим исходима</p>	
<p>оцена (5)</p>	<ul style="list-style-type: none"> • Пројектни задатак: Пројектни задатак је урађен изузетно квалитетно, са свим функционалностима правилно имплементираним. Код је оптимизован, документација је комплетна и добро организована. Показује способност креативног размишљања и иновативног решења проблема.
<p>оцена (4)</p>	<ul style="list-style-type: none"> • Пројектни задатак: Пројектни задатак је урађен са свим траженим функционалностима, али са мањим пропустима или недоследностима. Код је функционалан, али може бити мање оптимизован или са некомплетном документацијом. Показује солидно разумевање проблематике, али без већих иновација или креативних решења.
<p>оцена (3)</p>	<ul style="list-style-type: none"> • Пројектни задатак: Пројектни задатак је урађен, али садржи бројне грешке или је непотпун. Код је функционалан, али слабо оптимизован и документација може бити некомплетна. Показује основно разумевање проблема, али без дубине и са ограниченом креативношћу.
<p>оцена (2)</p>	<ul style="list-style-type: none"> • Пројектни задатак: Пројектни задатак је урађен, али садржи озбиљне недостатке, функционалности нису исправно имплементиране или су делимично урађене. Код је слабог квалитета, неоптимизован, и документација је минимална или изостављена. Ученик показује минимално разумевање и недостатак креативности.
<p>оцена (1)</p>	<ul style="list-style-type: none"> • Пројектни задатак: Пројектни задатак је непотпун или потпуно неисправан, са бројним функционалним недостацима. Код је лоше написан, без оптимизације и документације. Ученик показује врло слаб ниво разумевања и није способан да креативно решава проблеме.
<p>Напомена</p>	

Предметни професор:
Срђан Кушаковић

¹ Наставник додаје онолико табела колико има наставних
тема/модула/области