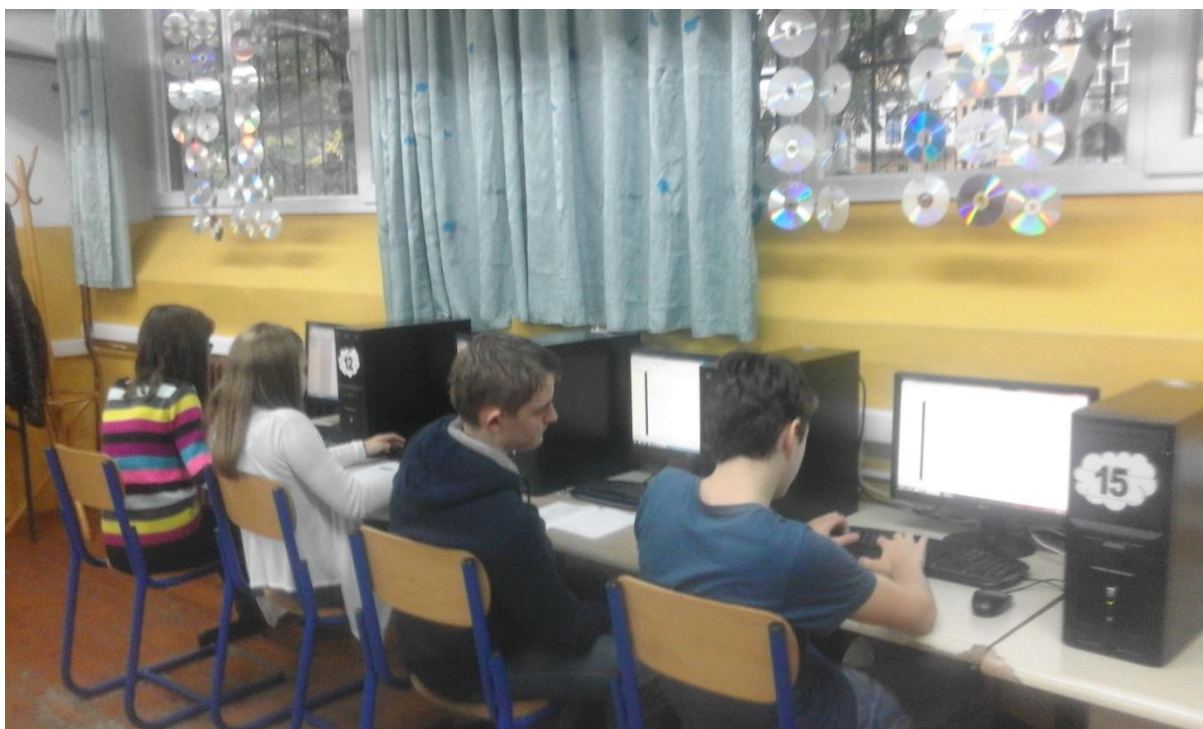


***ПИСАНА ПРИПРЕМА ЗА ОГЛЕДНИ ЧАС  
ИЗ ИНФОРМАТИКЕ***



***Бања Лука, 21.10.2015.***

***Професор информатике:***

***Душанка Томић***

<b>ОПШТИ ПОДАЦИ О ЧАСУ</b>	
НАЗИВ ШКОЛЕ:	ЈУ ОШ „Иво Андрић“
МЈЕСТО:	Бања Лука
РАЗРЕД И ОДЈЕЉЕЊЕ:	IX <sub>2</sub>
ДАТУМ:	21.10. 2015.
НАСТАВНИ ЧАС:	Пети час ( 17 <sup>15</sup> -18 <sup>00</sup> )
НАСТАВНИК ИНФОРМАТИКЕ:	Душанка Томић

<b>ОПШТИ МЕТОДИЧКИ ПОДАЦИ</b>	
НАСТАВНИ ПРЕДМЕТ:	Основи информатике
НАСТАВНА ТЕМА:	Програмирање
НАСТАВНА ЈЕДИНИЦА:	Рјешавање проблема примјеном низова
ТИП ЧАСА:	Вјежба

### *ИНОВАЦИЈА У НАСТАВИ – (оглед)*

Основне слабости и недостаци концепције традиционалне наставе се огледају у томе што се темељи на претпоставци о једнакости ученика. Сви ученици се поучавају истим темпом. Пракса показује да би у планирању, програмирању, припремању и реализацији наставе требало уважавати претходна знања и искуства ученика, опште и специјалне способности, њихове склоности, различите потребе и интересовања, особености њиховог памћења, начин њиховог реаговања, облике и стилове учења. Да би се то ефикасно постигло неопходно је примјењивати индивидуализовану наставу, која у складу са наведеним пружа могућност да се наставни садржаји функционално усвоје у складу са индивидуалним карактеристикама ученика.

Дакле, индивидуализована настава се темељи на индивидуалним разликама међу појединцима. Наставни захтјеви се усклађују са индивидуалним способностима и карактеристикама у развоју и раду ученика. Сваком ученику је омогућено да напредује својим темпом и према својим могућностима, тако да им то повећава мотивацију. Ученици самостално раде и уче, а њихов рад се континуирано прати и вреднује, јер одмах на часу добијају повратну информацију о свом напредовању. Посебна вриједност индивидуализације је у бољој искориштености менталног капацитета ученика. Свим ученицима је омогућен интелектуални напредак (Вилотијевић, 1999).

Како бих наставу информатике функционално прилагодила могућностима свих ученика одлучила сам се за примјену индивидуализоване наставе. Обзиром да постоји више облика индивидуализације наставног процеса, одлучила сам да на овом часу покажем *индивидуализацију наставног процеса из информатике, примјеном задатака на више нивоа сложености*. Примјена овог облика индивидуализације темељи се на нормалној дистрибуцији ученика у одјељењу према њиховим способностима и индивидуалним потребама, тако да у сваком одјељењу половина ученика постиже просјечне резултате, а друга половина ученика су надпросјечни и испод просјечни. У складу са могућностима ученика обрађују се два програмска језика C++ и Q-Basic.

За напредне ученике припремљени су задаци у C++-у, програмском језику који има напредне могућности. За просјечне ученике припремљени су задаци у Q-Basic-у, програмском језику ограничених могућности који је подесан за учење програмирања. Ученици са потешкоћама у учењу рјешаваће најједноставније задатке, такође примјеном Q-Basic програмског језика. Треба обратити пажњу, код примјене овог облика индивидуализације, да ученици који раде у Q-Basic програму, такође неће ићи истим

темпом при рјешавању задатака, дакле и овдје су припремљени додатни задаци за ученике који су напреднији.

Комплетан час осмишљен је тако да ученици задатке рјешавају директно у програмском едитору, дакле употребом рачунара. На овај начин ученици су више мотивисани за учење. Рад на рачунару им је забаван, па тако уче на лакши начин. Активност ученика на часу је већа јер сваки ученик има рачунар и труди се да заврши задатак.

## *МОТИВАЦИЈА*

Наставу најчешће прилагођавамо могућностима просјечног ученика, јер је таквих највише у одјељењу. Циљ је да ученицима који теже савладавају градиво (ученици са потешкоћама у учењу) прилагодим задатке тако да их успију ријешити и тако осјетити успјех и повећати своје самопоуздање, а ученике који брже напредују да рјешавањем тежих задатака више мисаоно ангажујем.

Веома је важно мотивисати ученике за учење. Приметила сам да су ученици који иду на додатну наставу више мотивисани и на редовном часу, ако задатке раде коришћењем програма C++. Њихова пажња, залагање и активност у рјешавању задатака у C++ су ми били подстицај да одређене наставне садржаје реализујем на тај начин. Самим тим прилагођавам наставу на часовима информатике, према потребама и жељама ученика. *Жеља ми је да сви ученици буду активни на часу информатике и да омогућим свим ученицима да напредују према својим могућностима.* Рјешавањем задатака у жељеном програму, ученицима ће бити занимљивије да самостално рјешавају задатке и покажу своје могућности. На овом часу ученицима желим и да пружим могућност избора, да сами процијене своје знање о програмирању и одаберу у ком програму задатке желе радити. Доношење те одлуке развија одговорност ученика за сопствене резултате. *Добијањем повратне информације ученици ће сами моћи да закључе на ком је нивоу њихово знање и да развију свијест о својим могућностима.*

*Знања која стекну на тај начин биће трајнија и функционалнија. Индивидуализацијом наставе кориштењем више програма на једном часу, наставни процес се модернизује, а ученици постају његови активни учесници*

## ПЛАНИРАЊЕ И ОРГАНИЗАЦИЈА ЧАСА

<i>УСАГЛАШЕНОСТ СА НПП- ом</i>	Наставним планом и програмом за IX разред основног образовања и васпитања из предмета информатика предвиђена је обрада наставне јединице „Рјешавање проблема примјеном низова“
<i>ЦИЉ ЧАСА</i>	Оспособљавање ученика за самостално рјешавање проблема, коришћењем низова.
<i>ЗАДАЦИ ЧАСА</i>	
а) Образовни:	<ul style="list-style-type: none"> <li>• Објаснити ученику значај примјене низова у програмирању;</li> <li>• Објаснити ученику значај програмирања</li> </ul>
б) Васпитни:	<ul style="list-style-type: none"> <li>• Развијање самосталности и одговорности у раду;</li> <li>• Развијање самопоуздања;</li> <li>• Развијање самокритичности и објективности у процјени властитог рада;</li> <li>• Развијање самопоуздање и повјерења у властите програмерске способности и интерес за програмирање;</li> <li>• Његовање такмичарског духа.</li> </ul>
в) Функционални:	<ul style="list-style-type: none"> <li>• Развијање аналитичког, логичког и процедуралног мишљења,</li> <li>• Развијати вјештина и способности у рјешавању проблемских задатака,</li> <li>• Развијати вјештину процјењивања најефикаснијег приступа рјешавању постављеног проблема ,</li> <li>• Развијати вјештину конкретне употребе ИКТ технологије и Интернета</li> <li>• Објаснити примјену програмирања у стварном животу</li> </ul>



## ТОК И САДРЖАЈ ЧАСА

### УВОДНИ ДИО ЧАСА (10 мин. )

На почетку часа укратко поновити опште знање о низовима, које смо учили прошли час. Постављати ученицима питања и од њих тражити одговор. Питања и одговори се налазе на презентацији, тако да наставник покреће презентацију и почиње са постављањем питања. Одговори су скривени, па тек када ученици одговоре или нико од њих не зна одговор наставник прикаже одговор. Затим се прелази на следеће питање.

1. П: Каког су типа сви подаци у низу и имају ли сви подаци исто име?  
О: Сви подаци у низу су истог типа и сви имају исто име.
2. П: Како приступамо подацима у једном таквом низу?  
О: Подацима у једном таквом низу приступамо помоћу цијелобројних индекса.
3. П: Који индекс има први елеменат у низу?  
О: Први елеменат у низу има индекс нула.
4. П: Можемо ли онда рећи којем математичком скупу припада скуп индекса једнодимензионалног низа?  
О: Скуп индекса једнодимензионалног низа припада скупу  $\mathbb{N}_0$ .
5. П: Како се декларише једнодимензионални низ?  
О: Једнодимензионални низ се декларише типом елемената, именом низа, те његовом дужином ( `tip_elementa ime_niza [dužina_polja]`).

## ГЛАВНИ ДИО ЧАСА ( 30 мин. )

Користи се иста презентација, која је већ покренута, па се приказује први задатак. Рјешења задатака су скривена. Напредни ученици раде задатке у C++, а остали те исте задатке раде у **Q-Basic** програму.

Док ученици рјешавају задатке ученик са потешкоћама у учењу добија наставни листић са задацима (задаци се прилагођавају могућностима ученика). Ученик је непрекидно под надзором наставника. Обзиром да има проблема са коришћењем тастатуре, наставник даје упутства и по потреби пружа подршку.

Ученици који су рјешили задатке јављају се подизањем руке, наставник прегледа задатке и у табелу уписује плус свим ученицима који су тачно ријешили (На листу папираје нацртана табела у којој се биљежи резултат сваког ученика. Она се на крају часа користи за оцијењивање. (Табела је дата у прилогу).

Када први ученик ријешити задатак, након што га наставник провјери и утврди да је тачан прикаже други задатак на презентацији.

Ученици задатак рјешавају 10-так минута, након чега наставник приказује слајд са рјешењем.

Наставник пролази кроз рјешење ред по ред, за ученике који нису успјели самостално да ријеше задатке. За ученике који не препишу све задатке у свеску, напоменути да ће се сва ријешења задатака налазити на мојој веб страници, тако да их могу преписати код куће. Моја веб страница је [dusankatomic.webs.com](http://dusankatomic.webs.com)

За ученике који тачно ураде све задатке са презентације припремљени су наставни листови са додатним задацима.

## ЗАДАЦИ :

### Задаци првог нивоа сложености (за ученика са потешкоћама у учењу)

Упутства са основним наредбама:

**REM**- Ту уписујемо КОМЕНТАР , нпр. Назив задатка

**CLS** – Очисти екран

**END** – Увијек стављамо на крају програма.

**PRINT**- се користи за испис текста или резултата на екрану. Стављаш **PRINT** тамо гдје желиш да ти се нешто испише на екрану.

**INPUT**– се користи да би корисник могао нешто уписати у твој програм. Дајеш кориснику да упише у извршни програм оно што се тражи у том програму

Наредба **IF**– Читамо “**ако је**”

**THEN** –Читамо “**ОНДА**”

нпр.,

**CLS**

**INPUT "Упиши број: ",x**

**IFx> 0 THENPRINT "Позитиван је”**

Читамо: АКОје х веће од нуле, ОНДАиспиши "Позитиван је”

Ако желиш да корисник унесе неко име или ријеч код промјенљиве ставиш знак \$, нпр:

**INPUT“ Unesinekoime” X\$**

Ако желиш да корисник унесе неки број онда ставиш само промјенљиву нпр.

**INPUT“ Unesinekibroј”, x**

Задаци:

1. Напиши програм који за два броја рачуна збир и разлику.

**Рјешење:**

REM: zbiri razlika dva broja

CLS

INPUT “ Unesi prvi broj”, a

INPUT “ Unesi drugi broj”, b

c= a + b

d = a - b

PRINT “ Zbir je “ , c

PRINT” Razlika je “, d

END



2. **Напиши програм који пита корисника, како се зове, а затим исписује поруку „ Дobar дан, \_\_\_\_\_ . ( У продужетку ће написати име које корисник унесе:**

**Рјешење::**

```
REM :Pozdrav korisniku
```

```
CLS
```

```
INPUT " Unesi svoje ime", X$
```

```
PRINT " Dobar dan, ", X$
```

```
END
```

3. **Напиши програм који рачуна површину правоугаоника**

```
REM : Povrsina pravougaonika
```

```
CLS
```

```
INPUT „Unesi stranicu a= ”, a
```

```
INPUT „Unesi stranicu b= ”, b
```

```
P = a * b
```

```
PRINT „Povrsina je“ , P
```

## Задаци другог и трећег нивоа сложености

(Раде их ученици у програмском језику QBasic или C++).

1. **Напиши програм који учитава низ од  $n$  бројева, а затим исписује чланове низа у обрнутом редослиједу .**

Испис на екрану:

Колико чланова: 6; Унеси чланове : 4,2,34,5,7,1 Испис: 1,7,5,34,2,4

QBasic	C++
<pre>REM :Obrnuti niz CLS INPUT "unesite broj clanova niza",n DIM x(n)  FOR i = 1 TO n   PRINT i; " ";   INPUT "x(i)"; NEXT i  FOR i = n TO 1 STEP -1   PRINT "clan("; i; ") = "; x(i) NEXT i END</pre>	<pre>#include &lt;cstdlib&gt; #include &lt;iostream&gt; using namespace std; int main() {   int n;   cout &lt;&lt; "Koliko ce clanova imat iniz? "   &lt;&lt;endl;   cin&gt;&gt; n;   int niz[n];   cout &lt;&lt; "Unos clanova niza: " &lt;&lt;endl;   for( i = 0; i&lt; n; i++){     cin&gt;&gt;niz[i];   }    cout&lt;&lt; "Ispis clanova niza u obrnutom redu: "   &lt;&lt;endl;   for( i = n; i&gt;=0; i--){     cout&lt;&lt;niz[i];   }   }   system("PAUSE");   return 0; }</pre>

**2. Напиши програм који читава низ од n бројева, а затим исписује само чланове који су парни.**

Испис на екрану:

Колико чланова: 5; Унеси чланове: 4, 2, 7, 34, 21 ; Члан : 4, 2, 34.

QBasic	C++
<pre> REM Parni clanovi niza CLS INPUT "unesi broj clanova niza",n DIM x(n) FOR i = 1 TO n   PRINT i; " ";   INPUT " " x(i) NEXT i FOR i = 1 TO n   IF x(i) MOD 2 = 0 THEN PRINT "clan("; i; ") = "; x(i) NEXT i END </pre>	<pre> #include &lt;cstdlib&gt; #include &lt;iostream&gt; using namespace std; intmain() {   int n;   cout &lt;&lt; "Unesitebrojclanovaniza:";   cin&gt;&gt; n;   inta[n];   cout &lt;&lt; "Unesiteclanoveniza:"&lt;&lt;endl;   for (inti=0; i&lt; n; i++){     cin&gt;&gt; a[i];   }   cout&lt;&lt; "Ispisparnihclanovaniza:" &lt;&lt;endl;   for (inti=0; i&lt; n; i++){     if(a[i] % 2 ==0){       cout&lt;&lt; a[i] &lt;&lt;endl;     }   }   system("PAUSE");   return 0; } </pre>

3. **Напиши програм који читава низ од n бројева, а затим исписује само оне који су већи од њихове аритметичке средине.**

Испис на екрану:

Колико бројева: 5 ; Унеси бројеве: 2,3,5,6,8 Излаз: Аритметичка средина. 4,8 Већи су: 5,6,8.

QBasic	C++
<pre> REM :veci od aritmeticke sredine CLS INPUT "N=", n DIM x (n) PRINT "unesite brojeve" s = 0 FOR i = 1 TO n INPUT x(i) s = s + x(i) NEXT i Asr = s / n PRINT "aritmeticka sredina je ", Asr PRINT "vecisu " FOR i = 1 TO n IF x(i) &gt;Asr THEN PRINT x(i) END IF NEXT i PRINT END </pre>	<pre> #include &lt;cstdlib&gt; #include &lt;iostream&gt; using namespace std;  int main() {     int n;     int br=0;     double s=0;     cout&lt;&lt; "Unesite broj clanova niza:";     cin&gt;&gt; n;     int a[n];     cout&lt;&lt; "Unesite clanove niza:"&lt;&lt;endl;     for (inti=0; i&lt; n; i++){         cout&lt;&lt;i;         cin&gt;&gt;a[i];     }     cout&lt;&lt; "----- &lt;&lt;endl;     for (inti=0; i&lt; n; i++)     {         s=s+a[i];         br=br+1;     }     double As=s/br;     for(inti=0;i&lt;n;i++) {="" if(a[i]==""&gt; As)     {         cout&lt;&lt;"a["&lt;&lt;i&lt;&lt;"]="&lt;&lt;a[i]&lt;&lt;endl;     }     }     cout&lt;&lt;"Aritmeticka sredina iznosi: "&lt;&lt;As&lt;&lt;endl;     system("PAUSE");     return 0; } </pre>

## Додатни задаци:

### За напредне ученике:

1. Напиши програм који учитава низ од  $n$  бројева, а затим исписује сусједне чланове чија је сума парна .

Испис на екрану:

Унеси број чланова низа:5, Унеси чланове: 3,2,4,1,9.

Резултат:  $x(2)+x(3)=6$

$x(4)+x(5)=10$

QBasic	C++
<pre>REM Susjedni članovi čiji je zbir paran CLS PRINT "unesi broj clanova niza" INPUT n DIM x(n)  FOR i = 1 TO n   PRINT i; "----&gt;";   INPUT x(i) NEXT i  FOR i = 1 TO n - 1   IF (x(i) + x(i + 1)) MOD 2 = 0 THEN   PRINT "rezultat"; "x("; i; ")";   "+"; "x("; i+1; ")="; x(i)+x(i+1) NEXT i  END</pre>	<pre>#include &lt;cstdlib&gt; #include &lt;iostream&gt;  using namespace std;  intmain() {   int n;   cout &lt;&lt; "Unesite broj clanova niza:";   cin&gt;&gt; n;   inta[n];   cout &lt;&lt; "Unesite clanove niza:"&lt;&lt;endl;   for (inti=0; i&lt; n; i++){     cin&gt;&gt; a[i];   }   cout&lt;&lt; "Ispis" &lt;&lt;endl;   for (inti=0; i&lt; n; i++){     if((a[i]+a[i+1]) % 2 ==0){       cout&lt;&lt; a[i] &lt;&lt;endl;     }   }   system("PAUSE");   return 0;</pre>

**2. Напиши програм који читава низ од n бројева, а затим их исписује у растућем редослиједу.**

Испис на екрану:

Колико бројева: 5 ; Унеси бројеве : 22, 7, 5, 4, 9

Сортирани низ је: 4, 5, 7, 9, 22

QBasic	C++
<pre> REM :RASTUĆI NIZ CLS INPUT „ Koliko brojeva“, n DIM x(n) PRINT „ Unesi brojeve“ FOR i= 1 TO n INPUT x(i) NEXT FOR i = 1 TO n-1 FOR j = i+1 TO n IF x(j)&lt;x(i) THEN SWAP x(i),x(j) END IF NEXT j NEXT i PRINT „ Sortirani niz je“; FOR i= 1 TO n PRINT x(i) PRINT END </pre>	<pre> #include&lt;cstdlib&gt; #include&lt;iostream&gt; usingnamespacestd; intmain() { inti,j; intn; cout&lt;&lt;„Unesi broj clanova niza „; cin&gt;&gt;n; intniz[n]; cout&lt;&lt;„Unesi elemente niza“&lt;&lt;endl; for (i=0;i&lt;n;i++) { cin&gt;&gt;niz[i]; } for (i=0;i&lt;n;i++) for (j=i+1;j&lt;n;j++) if (niz[i]&gt;niz[j]) swap (niz[i],niz[j]); cout&lt;&lt;„Niz u rastucem redoslijedu“&lt;&lt;endl; for (i=0;i&lt;n;i++) cout&lt;&lt;niz[i]&lt;&lt;endl; system („PAUSE“); return 0; </pre>

### ЗА ПРОСЈЕЧНЕ УЧЕНИКЕ:

(Они задатке раде само у Qbasic програму, па ће и рјешења бити у том програму.)

1. Написати програм који учитава низ од  $n$  бројева, а затим исписује сваки други члан унесеног низа

Испис на екрану:

Унеси колико желиш бројева : 6

Унеси бројеве : 2, 4, 6, 9, 10, 12.

То су следећи чланови:  $x(2) = 4$ ;  $x(4) = 9$ ;  $x(6) = 12$

#### QBasic

```
REM: svaki drugi broj iu nizu
CLS
INPUT „ Unesi koliko zelis brojeva“, n
DIM x(n)
FOR i = 1 TO n
INPUT „ Unesi brojeve“, x(i)
NEXT i
FOR i = 2 TO n STEP 2
PRINT „ To su sledeći članovi“; „x(„; i ; „)=“; x(i)
NEXT i
END
```

2. Написати програм који учитава низ од  $n$  бројева, а затим их исписује следећим редослиједом: „Позитивни чланови низа су:“, „Нуле су: „ , „ Негативни чланови низа“

Испис на екрану:

Број чланова: 5; Чланови: 9,-2, 3, -7,0; Испис: позитивни  $x(1)=9$ ,  $x(3)=3$ , ; нула  $x(5)=0$ , Негативни  $x(2)=-2$ ,  $x(4)=-7$ .

### QBasic

```
REM: pozitivni, nule I negativni članovi niza
CLS
PRINT "unesi broj članova niza"
INPUT n
DIM x(i)

FOR i = 1 TO n
  PRINT i; "---->"
  INPUT x(i)
NEXT i
PRINT "pozitivni članovi niza"
FOR i = 1 TO n
  IF x(i) > 0 THEN
    PRINT "x("; i; ") = "; x(i)
  END IF
NEXT i
PRINT "nule"
FOR i = 1 TO n
  IF x(i) = 0 THEN
    PRINT "x("; i; ") = "; x(i)
  END IF
NEXT i
PRINT "negativni članovi niza"
FOR i = 1 TO n
  IF x(i) < 0 THEN
    PRINT "x("; i; ") = "; x(i)
  END IF
NEXT i
END
```



### **ЗАВРШНИ ДИО (5 мин. )**

На крају часа анализирати колико је који ученик урадио задатака и оцијенити ученике који су били активни. Задатке који су припремљени као додатни задаци, а нису урађени на часу, дати ученицима за домаћу задаћу (задаци су припремљени на листићима). Ученицима са потешкоћама у учењу такође припремити задатке за задаћу у складу са њиховим могућностима.

## Додатак – (наставни листићи за ученике)

### Задаци за ученика са потешкоћама у учењу:

1. Наредба у Qbasic програму која служи да би корисник могао нешто уписати на екрану је (заокружи тачан одговор):
  - a) PRINT
  - б) INPUT
  - в) THEN
2. Наредба у Qbasic програму која служи за испис текста на екрану је (заокружи тачан одговор)::
  - a) PRINT
  - б) INPUT
  - в) REM
3. Наредба у Qbasic програму која служи да уписујемо коментар је (заокружи тачан одговор):
  - a) PRINT
  - б) REM
  - в) THEN

### Додатни задаци за други ново:

1. Написати програм који учитава низ од  $n$  бројева, а затим их исписује сваки други члан унесеног низа.

Испис на екрану:

Унеси колико желиш бројева : 6

Унеси бројеве : 2, 4, 6, 9, 10, 12.

То су следећи чланови:  $x(2) = 4$ ;  $x(4) = 9$ ;  $x(6) = 12$

2. Написати програм који учитава низ од  $n$  бројева, а затим их исписује следећим редослиједом: „Позитивни чланови низа су:“, „Нуле су: „ , „ Негативни чланови низа“

Испис на екрану:

Број чланова: 5; Чланови: 9,-2, 3, -7,0; Испис: позитивни  $x(1)=9$ ,  $x(3)=3$ , ; нула  $x(5)=0$ , Негативни  $x(2)=-2$ ,  $x(4)=-7$ .

### Додатни задаци за трећи ново:

1. **Напиши програм који учитава низ од  $n$  бројева, а затим исписује сусједне чланове чија је сума парна .**

Испис на екрану:

Унеси број чланова низа: 3,2,4,1,9.

Резултат:  $x(2)+x(3)=6$

$x(4)+x(5)=10$

2. **Напиши програм који учитава низ од  $n$  бројева, а затим их исписује у растућем редослиједу.**

Испис на екрану:

Колико бројева: 5 ; унеси бројеве : 22, 7, 5, 4, 9

Сортирани низ је: 4, 5, 7, 9, 22

**Табела за праћење рада ученика:**

<b>Р. бр.</b>	<b>Име и презиме</b>	<b>Прогр. језик</b>	<b>Задатак 1</b>	<b>Задатак 2</b>	<b>Задатак 3</b>	<b>Додатни задатак 1</b>	<b>Додатни задатак 2</b>
1.	<b>Анабела Арар</b>						
2.	<b>Анастасија Арар</b>						
3.	<b>Игор Бујак</b>						
4.	<b>Алекса Вуцеља</b>						
5.	<b>Вања Добрњац</b>						
6.	<b>Исидора Драгичевић</b>						
7.	<b>Кристина Симић</b>						
8.	<b>Тамара Петрушић</b>						
11.	<b>Никола Кнежевић</b>						
12.	<b>Габријела Сужњевић</b>						
13.	<b>Антонио Мендеш</b>						
14.	<b>Страхиња Максимовић</b>						