

III. Tétel (30 pont)

Az 1-es pontban írástok a vizsgalapra a helyes válasz betűjelét.

1. Képezzük az összes 5 betűs karaktersort a $\{p, r, o, b, a\}$ halmaz elemeiből. A megoldási módszer melyik algoritmussal egyenértékű? (4p.)
- a. variációk generálása b. permutációk generálása c. Részhalmazok generálása d. Descartes szorzat generálása

Az alábbi feladatok megoldásait írástok a vizsgalapra.

2. Tekintsük a mellékelt rekurzív alprogramot. Mit ír ki a képernyőre az `f1(4)`; meghívása? (6p.)
- ```
procedure f1(x:integer);
begin
 if x<=9 then
 begin
 write(x+1);
 f1(x+2)
 end
end;
```
3. Adjátok meg a **ordonare** alprogram teljes leírását, amelynek paraméterei: az **x**, legtöbb 100 valós számból álló egydimenziós tömb és az **n** ( $n \leq 100$ ) természetes szám, amely **x** tömb elemeinek száma. Az alprogram írja ki azt az egydimenziós tömböt, amelyet úgy kapunk, hogy az eredeti tömb páratlan értékű elemeit rendezzük egymás között növekvő sorrendben, a páros értékű elemek helyben maradnak. A számokat egymástól egy-egy szóközzel elválasztva kell kiíratni. (10p.)
- Példa:** ha  $n=6$  és  $x=(7, 11, 2, -8, -3, 10)$ ,  
akkor az alprogram kiírja: `-3 7 2 -8 11 10`
4. A **numere.txt** szöveges állomány első sorában egy **n** ( $1 \leq n \leq 100$ ) természetes szám található, a következő **n** sorban, pedig az **x, y** ( $-100 \leq x \leq y \leq 100$ ) egész számpárok, amelyek az **ox** koordinátatengelyen felvett **[x, y]** szakaszok végpontjait jelentik.
- a) Írástok a memória és a futási idő szempontjából hatékony **Pascal** programot, amely kiolvassa az állományból az adatokat, és meghatározza az **n** szakasz metszetét, majd szóközzel elválasztva kiírja a képernyőre a kért szakasz két végpontját, illetve 0-t ha a szakaszoknak nincs közös pontjuk. (6p.)
- b) Írástok le a saját szavaitokkal az általatok használt megoldási módszert és magyarázzátok meg, hogy miben rejlik a hatékonysága. (4p.)
- Példa:** ha a **numere.txt** a mellékelt tartalommal rendelkezik, akkor a képernyőn megjelenik a:
- 3 5

```
5
-7 10
3 20
-5 5
0 12
-8 30
```