

Dynamické pole celých čísel

predpokladáme deklaráciu:

```
type TPole = array of integer;
```

1. Napíšte funkciu `nahodne(n:integer):TPole`, ktorá vygeneruje **n**-prvkové pole s náhodnými číslami od 0 do 99.
2. Napíšte funkciu `vyrobPole(p:array of integer):TPole`, ktorá dostane parameter *otvorené pole* a z neho vyrobí dynamické pole.
3. Napíšte funkciu `toStr`, ktorá dostane parameter otvorené pole a vytvorí reťazec, v ktorom budú prvky poľa oddelené medzerami.
4. Napíšte funkciu `sucet`, ktorá dostane parameter otvorené pole a vráti súčet jeho prvkov.
5. Napíšte procedúru `pridaj(x:integer; var p:TPole)`, ktorá zmení pole `p` tak, že na jeho koniec pridá nový prvok `x`.
6. Predpokladáme, že pole je vzostupne utriedené. Napíšte procedúru `zarad(x:integer; var p:TPole)`, ktorá pridá nový prvok do poľa `p` tak, aby ostalo utriedené.
7. Napíšte funkciu `tried(p:array of integer):TPole`, ktorá pomocou procedúry `zarad` utriedi pole čísel.
8. Napíšte procedúru `vyhodDupl(var p:TPole)`, ktorá z dynamického utriedeného poľa vyhodí všetky duplikáty - pole sa možno skrúti.
9. Napíšte funkciu `spoj(a,b:array of integer):TPole`, ktorá dve polia sa “zreťazí” do jedného.
10. Napíšte funkciu `delitele(n:integer):TPole`, ktorá vráti dynamické pole, ktoré obsahuje všetkých deliteľov daného čísla okrem samého seba. Môžete použiť procedúru `pridaj`.

Dynamické pole dvojíc reťazcov – slovník

Predpokladáme deklaráciu:

```
type
```

```
  TZaznam = record a,b:string end; //a reprezentuje anglické, b slovenské slovo  
  TPole = array of TZaznam;      // slovník
```

11. Napíšte funkciu `zaznam(aa,bb:string):TZaznam`, ktorá vytvorí záznam z dvoch reťazcov `aa` a `bb`.
12. Napíšte funkciu `function vyrob(p:array of TZaznam):TPole`, ktorá z daného vstupného poľa vytvorí dynamické pole.
13. Napíšte funkciu `toStr(p:array of TZaznam):string`, ktorá vytvorí z poľa reťazec - riadky sú oddelené `#13#10`.
14. Napíšte funkciu `poz(x:string; p:array of TZaznam):integer`, ktorá vráti pozíciu reťazca `x` v poli `p`. Parameter `x` reprezentuje anglické slovo, teda porovnávať budete časť `a` záznamu. Ak reťazec nenájde, vráti hodnotu `-1`.
15. Napíšte procedúru `pridaj(x:TZaznam; var p:TPole)`, ktorá pridá záznam `x` na koniec poľa.
16. Napíšte procedúru `vyhod(x:string; var p:TPole)`, ktorá vyhodí z poľa `p` záznam, ktorého anglické slovo (časť `a`) sa rovná reťazcu `x`. Vyhadzujte tak, že sa na miesto vyhodeneho dáte posledný prvok poľa a pole skrútime o 1 prvok. Využite funkciu `poz`.
17. Napíšte procedúru `zapis(subor:string; p:array of TZaznam)`, ktorá zapíše pole do súboru tak, že do každého riadku dá jeden záznam, časti `a` a `b` záznamu oddelíte bodkočiarkou. Napr.
`dog;pes`
`ball;lopta`
18. Napíšte procedúru `citaj(subor:string; var p:TPole)`, ktorá prečíta pole zo súboru formátu vytvorenom v predchádzajúcej úlohe.

Dvojrozmerné dynamické pole

Predpokladáme deklaráciu:

```
type TPole2 = array of array of integer;
```

19. Napíšte procedúru `pridaj(x:array of integer; var p:TPole2)`, ktorá do dvojrozmerného poľa čísel na koniec pridá nový riadok, napr.

```
var
  p:TPole2;
begin
  p:=nil;
  pridaj([1,2,3,4,5],p);
  pridaj([],p);
  pridaj([6,7,8],p);
```

20. Napíšte funkciu `toStr(p:TPole2):string`, ktorá vytvorí s dvojrozmerného poľa reťazec tak, že prvky v jednotlivých riadkoch budú oddelené medzerami a riadky budú oddelené znakmi `#13#10`.

21. Napíšte procedúru `zapis(subor:string; p:TPole2)`, ktorá zapíše dvojrozmerné pole (podobným spôsobom ako v predchádzajúcom príklade) do súboru.

22. Napíšte procedúru `citaj(subor:string; var p:TPole2)`, ktorá vytvorí dvojrozmerné pole čísel so súboru (súbor vyzerá tak, ako ste ho vytvorili v predchádzajúcom príklade).