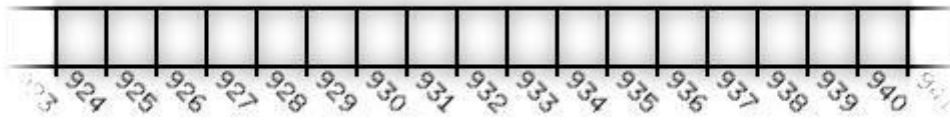


УКАЗАТЕЛИ

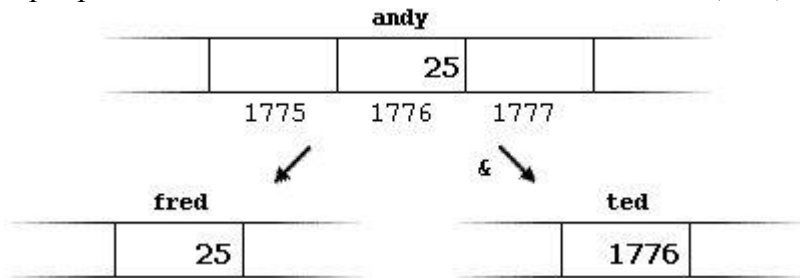
Паметта на компютъра – последователност от номерирани клетки с големина 1 байт



Всеки байт може да бъде намерен по неговия номер (адрес)

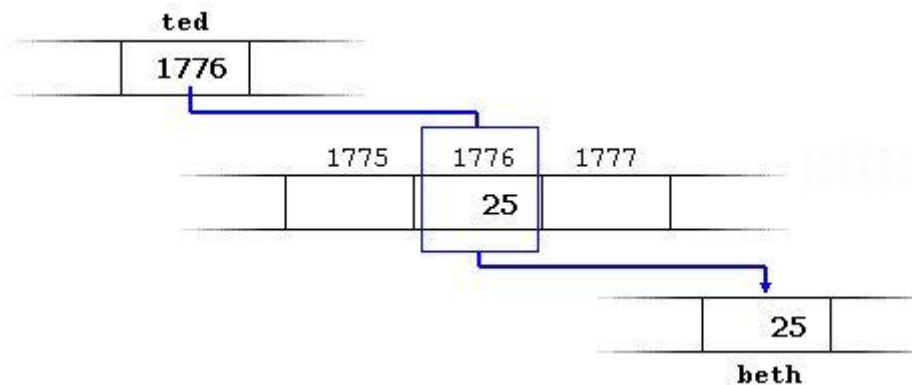
Операция намиране на адрес & (адресация, Reference)

char andy; резервира място в паметта за едно "char" число sizeof(char) байта
 andy=25;
 fred=andy;
 ted=&andy;
 (ted=адреса на andy)



Операция намиране на съдържанието по адрес * (де-адресация, Dereference)

andy=25;
 fred=andy;
 ted=&andy;
 beth=*ted;
 (beth =стойността от
 адреса сочен от ted)



Деклариране на променливи указатели:

type *ident; декларира 1) че е указател 2) тип на указваната променлива

инициализация: char c, *pc=&c; => char c; char *pc; pc=&c;

<pre>#include <stdio.h> int main() { char c,*pc=&c; short s,*ps=&s; int k,*pk=&k; long m,*pm=&m;float f,*pf=&f; double d,*pd=&d; printf ("char: %d %d\n",sizeof(*pc),sizeof(pc)); printf ("short: %d %d\n",sizeof(*ps),sizeof(ps)); printf ("int: %d %d\n",sizeof(*pk),sizeof(pk)); printf ("long: %d %d\n",sizeof(*pm),sizeof(pm)); printf ("float: %d %d\n",sizeof(*pf),sizeof(pf)); printf ("double: %d %d\n",sizeof(*pd),sizeof(pd)); printf ("pts:%x %x %x %x %x %x\n",pc,ps,pk,pm,pf,pd); return 0; }</pre>	<pre>char: 1 4 short: 2 4 int: 4 4 long: 4 4 float: 4 4 double: 8 4 pts:23ff77 23ff6e 23ff64 23ff5c 23ff54 23ff48</pre>
---	---

int *p1,p2; p1 - указател към int; p2 – int

Null pointer int *p=NULL; int *p=0; не указва никакъв адрес

void *p; Специален указател, който няма тип на указваната променлива

адресна аритметика

Указателите всъщност представляват цели числа указващи адреси от паметта. С тях може да се извършват определени операции:

++(Инкрементиране)и -- (декрементиране)

```
#include <stdio.h>
```

```
int main(){
```

```
char *mychar=(char *)1000;
```

```
short *myshort=(short *)2000;
```

```
long *mylong=(long *)3000;
```

```
mychar++; myshort++; mylong++;
```

```
printf("mychar %d\n", mychar);
```

```
printf ("myshort %d\n",myshort);
```

```
printf ("mylong %d\n",mylong);
```

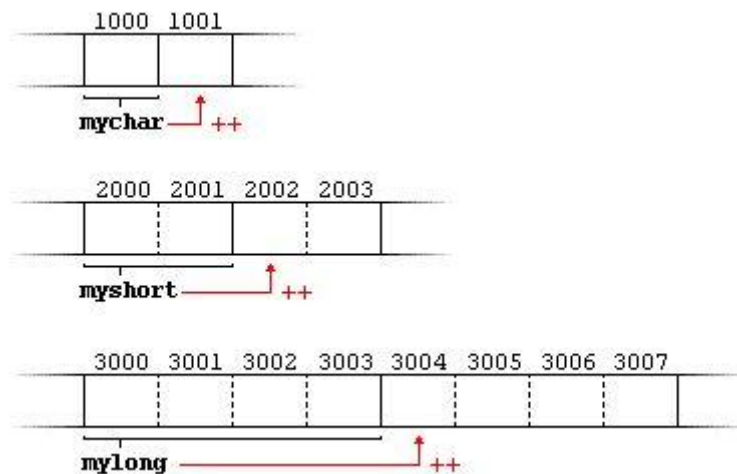
```
return 0;
```

```
}
```

mychar 1001

myshort 2002

mylong 3004



Сума на указател с цяло число (положително или отрицателно)

Разлика между 2 указателя

МАСИВИ

статична подредена колекция от елементи от еднакъв тип с равнопоставен достъп чрез изчисляем(и) индекс(и) до всеки елемент

В С липсва такъв тип, но съществува заместващ механизъм

```
int number[20], *p=number;
```

1) запазва място за 20 последователни елемента - int числа

2) number представлява **константен указател** към първия елемент

достъп до елемент на масива number[3], p[3], *(number+3), *(p+3);

първия елемент е *number, *(number+0), number[0]

p=number+4; *p – стойността на 5-я елемент на масива

Възможно ли е и какво означава number[-5] ? 3[number] ?