

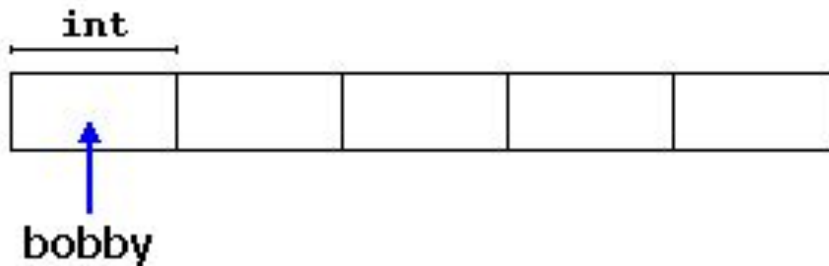
Динамическая память

- Представим случай, когда пользователю нужно указать объем памяти при выполнении программы.
- Нам потребуется выделять память динамический
- Для этого в C++ есть операторы `new` и `delete`

Операторы new и new[]

- указатель = new тип
- указатель = new тип [количество элементов]

```
int * bobby;  
bobby = new int [5];
```



- Так как ресурсы компьютера ограничен, иногда могут возникнуть проблемы выделения памяти. В C++ для этого есть два подхода решения:
 - Исключение `bad_alloc`
 - Специальный объект `nothrow`

```
bobby = new (nothrow) int [5];
```

Операторы delete и delete[]

- *delete* pointer;
- *delete []* pointer;

```

// rememb-o-matic
#include <iostream>
#include <new>
using namespace std;

int main ()
{
    int i,n;
    int * p;
    cout << "How many numbers would you like to type? ";
    cin >> i;
    p= new (nothrow) int[i];
    if (p == 0)
        cout << "Error: memory could not be allocated";
    else
    {
        for (n=0; n<i; n++)
        {
            cout << "Enter number: ";
            cin >> p[n];
        }
        cout << "You have entered: ";
        for (n=0; n<i; n++)
            cout << p[n] << ", ";
        delete[] p;
    }
    return 0;
}

```

```

How many numbers would you like to type? 1
Enter number : 75
Enter number : 436
Enter number : 1067
Enter number : 8
Enter number : 32
You have entered: 75, 436, 1067, 8, 32

```

Динамическая память и двумерные массивы

- `int **p = new int*[10];`
- `for(int i=0; i<10; i++)`
- `p[i] = new int[5];`

- `for(int i=0; i<10; i++)`
- `for(int j=0; j<5; j++)`
- `p[i][j]=1;`

Задания

- Перевести все задания на одномерные и двумерные массивы на динамические