BBM 201 Data Structures
Hacettepe University

# Lecture 4:
# Review of Dynamic Memory: Realloc

Resource:
Lecture Videos
www.mycodeschool.com/videos/pointers-and-arrays

# Realloc

*To modify the size of a memory block (to double, to reduce the size etc.), we use REALLOC.*

```c
int main()
{
    int n;
    printf("Enter size of array\n");
    scanf("%d",&n);
    int *A = (int*)malloc(n*sizeof(int)); //dynamically allocated array
    for(int i =0;i<n;i++)
    {
        A[i] = i+1;
    }
    int *B = (int*)realloc(A, 2*n*sizeof(int));
    for(int i = 0;i<n;i++)
    {
        printf("%d ",A[i]);
    }
}
```

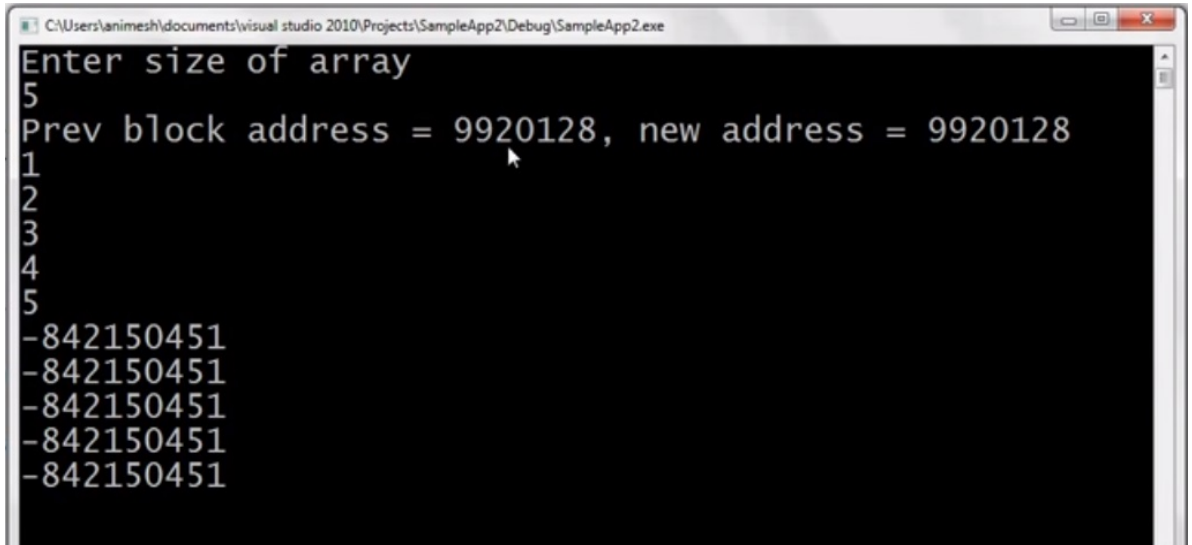
- *Realloc, extends the old memory if possible. Otherwise, it finds a new memory block to copy the content of A.*
- *As input, we pass the pointer A and the size of the new memory block.*

# Double the array size

```c
int main()
{
    int n;
    printf("Enter size of array\n");
    scanf("%d",&n);
    int *A = (int*)malloc(n*sizeof(int)); //dynamically allocated array
    for(int i =0;i<n;i++)
    {
        A[i] = i+1;
    }
    int *B = (int*)realloc(A, 2*n*sizeof(int));
    printf("Prev block address = %d, new address = %d\n",A,B);
    for(int i = 0;i<2*n;i++)
    {
        printf("%d\n",B[i]);
    }
}
```

# Double the array size

```c
int main()
{
    int n;
    printf("Enter size of array\n");
    scanf("%d",&n);
    int *A = (int*)malloc(n*sizeof(int)); //dynamically allocated array
    for(int i =0;i<n;i++)
    {
        A[i] = i+1;
    }
    int *B = (int*)realloc(A, 2*n*sizeof(int));
    printf("Prev block address = %d, new address = %d\n",A,B);
    for(int i = 0;i<2*n;i++)
    {
        printf("%d\n",B[i]);
    }
}
```



# Reduce the array size by half

```c
int main()
{
    int n;
    printf("Enter size of array\n");
    scanf("%d",&n);
    int *A = (int*)malloc(n*sizeof(int)); //dynamically allocated array
    for(int i =0;i<n;i++)
    {
        A[i] = i+1;
    }
    int *B = (int*)realloc(A, (n/2)*sizeof(int));
    printf("Prev block address = %d, new address = %d\n",A,B);
    for(int i = 0;i<n;i++)
    {
        printf("%d\n",B[i]);
    }
}
```

# Reduce the array size by half

```c
int main()
{
    int n;
    printf("Enter size of array\n");
    scanf("%d",&n);
    int *A = (int*)malloc(n*sizeof(int)); //dynamically allocated array
    for(int i =0;i<n;i++)
    {
        A[i] = i+1;
    }
    int *B = (int*)realloc(A, (n/2)*sizeof(int));
    printf("Prev block address = %d, new address = %d\n",A,B);
    for(int i = 0;i<n;i++)
    {
        printf("%d\n",B[i]);
    }
}
```