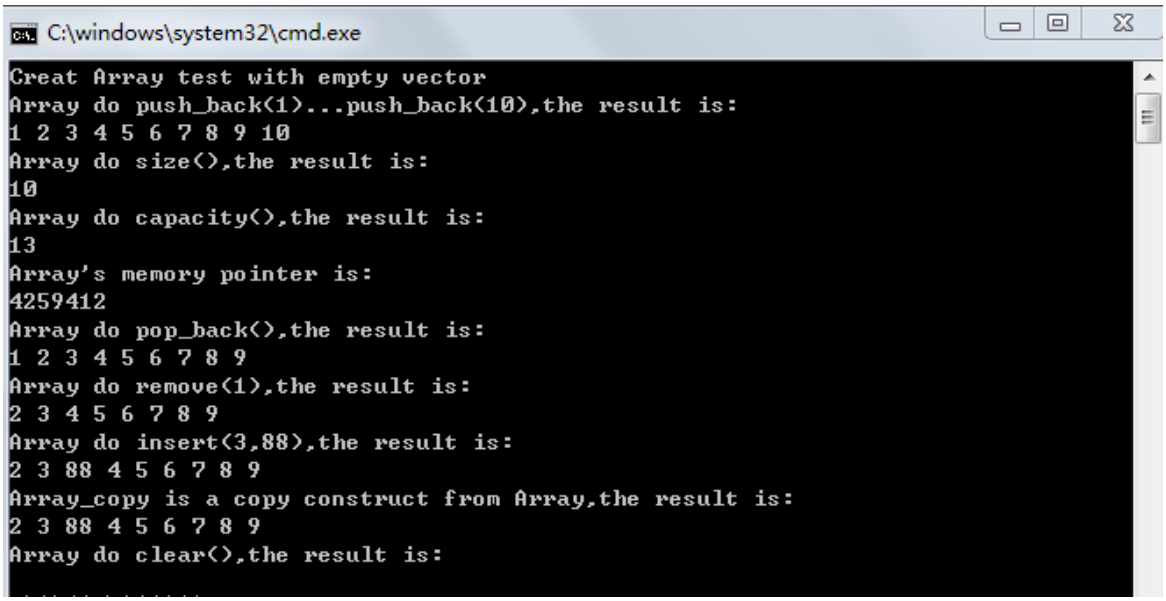
Report of Assignment 5

——Array

Meng Wei

Results:

When I define the class array, and give some tests. The results are as follows:



PS: Codes as follows:

```
// array1.cpp : Defines the entry point for the console application.
//
#include "stdafx.h"
#include<iostream>
#include<vector>
#include <malloc.h>
using namespace std;

class Array
{
public:
    vector<int> integers;
    int intnum;
    int data;
    Array() //default constructor
    {
        intnum = 0;
```

```
    data = int(&integers);
}
Array(const Array & copyvar) // copy constructor
{
    intnum = copyvar.intnum;
    integers = copyvar.integers;
}
Array& operator=(const Array & assignoperator) // assignment operator
{
    // check assignment to self
    if (this == &assignoperator)
        return(*this);

    delete[] &integers;//delete old memory
    intnum = assignoperator.intnum;
    integers = assignoperator.integers;

    return(*this);
}
~Array() // destructor
{
}
int cap()//define cap function
{
    int capa;
    int *data;
    capa = size_t (*data);
    return capa;
}
void push_back(int i)// define push_back function
{
    integers.push_back(i);
    intnum++;
}

void pop_back()//define pop_back function
{
    integers.pop_back();
    intnum--;
}

void remove(int i)//define remove function
{
    intnum--;
```

```
        integers.erase(integers.begin() + i - 1);
    }

    void insert( int i, int number)//define insert function
    {
        intnum++;
        integers.insert(integers.begin()+i-1,number);
    }

    int capacity()//define capacity function
    {
        return integers.capacity();
    }
    int size()//define size function
    {
        return intnum;
    }
    void clear()//define clear function
    {
        integers.clear();
        intnum = 0;
    }
};


int main()
{
    Array test;
    //---------------------------------
    cout << "Creat Array test with empty vector" << endl;
    //---------------------------------
    for (int i = 1; i <= 10; i++)
        test.push_back(i);
    cout << "Array do push_back(1)...push_back(10),the result is:" << endl;
    for (int i = 0; i < 10; i++)
        cout << test.integers[i] << ' ';
    cout << ' ' << endl;
    //---------------------------------
    cout << "Array do size(),the result is:" << endl;
    cout << test.size() << endl;
    //---------------------------------
    cout << "Array do capacity(),the result is:" << endl;
    cout << test.capacity() << endl;
    //---------------------------------
```

```
    cout << "Array's memory pointer is:" << endl;
    cout << test.data << endl;
    //--------------------------------
    test.pop_back();
    cout << "Array do pop_back(),the result is:" << endl;
    for (int i = 0; i < test.intnum; i++)
        cout << test.integers[i] << ' ';
    cout << ' ' << endl;
    //--------------------------------
    test.remove(1);
    cout << "Array do remove(1),the result is:" << endl;
    for (int i = 0; i < test.intnum; i++)
        cout << test.integers[i] << ' ';
    cout << ' ' << endl;
    //--------------------------------
    test.insert(3,88);
    cout << "Array do insert(3,88),the result is:" << endl;
    for (int i = 0; i < test.intnum; i++)
        cout << test.integers[i] << ' ';
    cout << ' ' << endl;
    //--------------------------------
    Array arraycopy(test);
    cout << "Array_copy is a copy construct from Array,the result is:" << endl;
    for (int i = 0; i < arraycopy.intnum; i++)
        cout << arraycopy.integers[i] << ' ';
    cout << ' ' << endl;
    //--------------------------------
    test.clear();
    cout << "Array do clear(),the result is:" << endl;
    for (int i = 0; i < test.intnum; i++)
        cout << test.integers[i] << ' ';
    cout << ' ' << endl;
    //--------------------------------

    return 0;
}
```