The "signature" of a function is the first line which gives the return type, the name of the function and the parameters with their types. As an example the signature of the gcd function is

```cpp
int gcd(int m, int n)
```

The function `multiprint` has the following signature.

```cpp
void multiprint(char c, int n)
```

The function is required to print n copies of the character `c`. Write the function.

There will invisible code which tests the function. It will call `multiprint` with certain values and print the result.

### Sample Test Cases

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Case 1</td>
<td>@ 0</td>
<td></td>
</tr>
<tr>
<td>Test Case 2</td>
<td>$ 7</td>
<td>$$$$$$$</td>
</tr>
<tr>
<td>Test Case 3</td>
<td>. 3</td>
<td>...</td>
</tr>
<tr>
<td>Test Case 4</td>
<td>* 5</td>
<td>*****</td>
</tr>
</tbody>
</table>
The due date for submitting this assignment has passed.
As per our records you have not submitted this assignment.

Sample solutions (Provided by instructor)

```cpp
#include <iostream>
#define repeat(x) for(int _iterator_i = 0, _iterator_limit = x; _iterator_i < _iterator_limit; _iterator_i++)
#define main_program int main()
#include <cmath>
using namespace std;

void multiprint(char c, int n){
    repeat(n) cout << c;
}

main_program{
    char c; cin >> c;
    int n; cin >> n;
    multiprint(c,n);
}
```