Lab 5.4.8.1 Memory on demand

Objectives

Familiarize the student with:

- Memory allocation
- *for* loops
- The modulo operator
- Printing on screen

Scenario

Write a program that allocates memory of a size requested by the user. This program requests a number from the user and checks if that number is less than 1 MB (1024*1024). If it's not, then the program prints the message. "Too much memory requested.". In the next step, the program allocates a one-dimesional array of characters (char) and fills this array with characters from "A" to "Z" (the 1st element (index 0) contains "A", the 26th element(index 25) contains "Z", the 27th element (index 26) contains "A" and so on). Then, the program prints the first 400 bytes of an array (or the whole array if it's smaller than 400 characters), 40 characters in each row. To simplify the program, you can use the *break* or *continue* statements. Remember to free up the array memory at the end of the program. Your version of the program must print the same result as the expected output.

```c
#include <stdio.h>

int main()
{
 /* your code */
 return 0;
}
```

Example input

```
100
```

Example output

```
ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMN
OPQRSTUVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZAB
CDEFGHIJKLMNOPQRSTUV
```

Example input

```
500
```

Example output

```
ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMN
OPQRSTUVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZAB
CDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOP
QRSTUVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZABCD
EFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQR
STUVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZABCDEF
GHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQRST
UVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGH
IJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQRSTUV
WXYZABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJ
```

Example input

```
1200500
```

Example output

```
Too much memory requested.
```

