**２年 プログラミング基礎＆演習I（2024年5月22日（水））**

1.　分割コンパイル

　これまで1つのファイルに1つのプログラムを記述してきた．一方，関数だけ別のプログラムファイルに記述して，いくつかのファイルを統合して実行ファイルを作成することができる．これを分割コンパイルという．次のプログラムを見てみよう．star\_all.cではキーボードから入力した数値だけ「\*」が階段状に表示されるプログラムである．

star\_all.c

#include <stdio.h>

void star (int num);

void star (int num)

{

int i;

i = 0;

while (i<num) {

printf("\*");

i++;

}

printf("\n");

}

int main(void)

{

int num;

printf("num=");

scanf("%d", &num);

star(num);

return 0;

}

このプログラムを2つのファイルに分割する．

star.c

#include <stdio.h>

void star (int num)

{

int i;

i = 0;

while (i<num) {

printf("\*");

i++;

}

printf("\n");

}

star\_main.c

#include <stdio.h>

void star (int num);

int main(void)

{

int num;

printf("num=");

scanf("%d", &num);

star(num);

return 0;

}

コンパイルは以下のように行う．

$ gcc star.c star\_main.c

あるいは，次のようにしてオブジェクトファイルを明示的に作成して，実行ファイルを作ることもできる．

$ gcc –c star.c (star.oというオブジェクトファイルが生成される)

$ gcc star.o star\_main.c

重要な点は，star.cのオブジェクトファイルを作っておくと，star.cの内容を変更しない限り，star.oを再度作り直す必要はない．例えば，star.cの内容は変更せず，star\_main.cの内容だけを変更した場合，以下のようにコンパイルを行う．

$ gcc star.o star\_main.c

　分割コンパイルが出来るからといって，1関数1ファイルにする必要はない．むしろ，ファイル数が増えて，プログラム全体の可読性が悪くなる．一方，1ファイルで1000行を超えるようなプログラムだと，非常に見渡しが悪くなる．適切にファイル分割を行う必要がある．

**第8回課題**

課題番号 No0522\_1.c**（要チェック）**

|  |
| --- |
| オブジェクトファイルstar\_space.oは，以下の仕様の関数が組み込まれている．  宣言：void star\_space(int num)  引数：num　正の整数値  　　　機能：num-1の空白を出力した後，「\*」を表示する．  　　　戻り値：なし  　入力された数値だけ以下の階段状になるようなプログラムを作成しなさい．ただし，main関数で，関数star\_spaceを用いたプログラムNo0522\_1.cを作成しなさい．  例えば，5を入力した場合，  \*  \*  \*  \*  \*  と表示される． |