

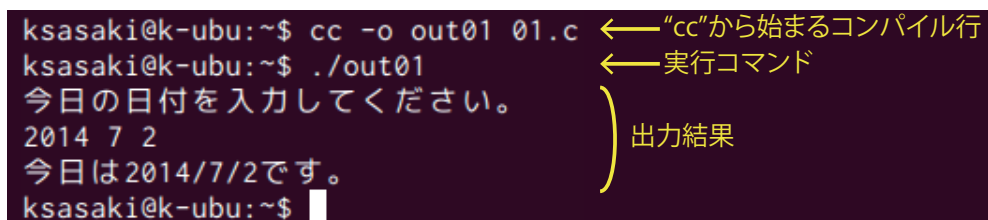
プログラミング IB 提出課題

第 1 回

以下の各課題について、ソースコード（.c ファイル）と実行結果のスクリーンショットを提出してください。
提出は **manaba**（ <https://manaba.tsukuba.ac.jp/ct/home> ）を通じて行います。manaba のコース一覧から [プログラミング IB] → [レポート] を選択し、次の注意事項に従って提出してください。

【注意事項】

1. 課題は必ず **Ubuntu** で実行してください。
2. 提出するファイル名は“(2桁の課題番号)”とします。
例えば、課題 01 についてのファイル名は、ソースコードは “01.c”、スクリーンショットは “01.png（.jpg でも可）” になります。
3. スクリーンショットは、“cc” から始まるコンパイルを行った行と、その後の実行コマンド、そして出力結果の 3 点がすべて分かるよう撮影してください。次の内容が撮影されていれば OK です。



```
ksasaki@k-ubu:~$ cc -o out01 01.c
ksasaki@k-ubu:~$ ./out01
今日の日付を入力してください。
2014 7 2
今日は2014/7/2です。
ksasaki@k-ubu:~$
```

4. それぞれのファイルは、manaba のレポート画面にある本日の課題からアップロードします。[ファイルを選択] からソースコードおよびスクリーンショットのファイルを選択し、[アップロード] をクリックしてください。課題が 4 問ある場合は計 8 つのファイルを提出することになりますので、この操作を 8 回繰り返してください。
また、最後に必ず「提出」ボタンを押してください。これを忘れると提出が未完了のままになります。
5. 提出期限は **10 月 4 日 23:59（明日いっぱい）** までです。また期限内であれば何度でも課題の再提出は可能ですが、再提出のたびに、必ず必要なファイル全てをアップロードし直してください。また期限を過ぎたものは、いかなる理由があっても提出を認めません。

【スクリーンショットの撮り方】

Ubuntu では、次の 2 通りの方法でスクリーンショットを撮影することができます。

1. キーボード右奥にある [PrtScn] キーを押すと、スクリーンショットダイアログが表示されるので、ファイル名と保存場所を指定して保存してください。
2. 画面左上の「Dash ホーム」で「スクリーンショット」と検索すると出てくるスクリーンショットアプリを使用しても撮影ができます。この場合、スクリーンショットの取得で「現在のウィンドウ」もしくは「取得する領域の選択」を選択し、必要な部分だけ撮影してもらっても結構です。

問 01

次の実行結果のように、身長 [m] と体重 [kg] を入力すると、BMI を計算して出力するプログラムを書きなさい。ただし以下の条件を満たすこと。

- (1) 入力値は main 関数内で受け取ること
- (2) BMI の計算を行う calc 関数を定義し、入力値を引数として渡すこと。
- (3) calc 関数は戻り値として、算出した BMI を main 関数に渡し、最終的な計算結果の表示は main 関数で行うこと
- (4) BMI の計算には以下の式を用いること

$$BMI = \text{体重 [kg]} / (\text{身長 [m]})^2$$

```
parallels@TA-Machine:~$ gcc -o 01 01.c
parallels@TA-Machine:~$ ./01
身長a [m]と体重b [kg]を入力してください
1.70
60
BMIは 20.761246 です。
parallels@TA-Machine:~$
```

問 02

次の実行結果のように、整数値を入力して、その 2 乗の値を求めるプログラムを書きなさい。ただし以下の条件を満たすこと。

- (1) 整数値は main 関数内で受け取ること
- (2) 整数値を 2 乗する関数 square を定義し、演算は square 関数内で行うこと。ただし、関数 square は引数を受け取らず、また戻り値もない関数とする。
- (3) 最終的な計算結果の表示は main 関数が行うこと。

```
parallels@TA-Machine:~$ gcc -o 02 02.c
parallels@TA-Machine:~$ ./02
整数値を入力してください：3
2乗すると9です。
parallels@TA-Machine:~$ ./02
整数値を入力してください：4
2乗すると16です。
parallels@TA-Machine:~$
```

問 03

次の実行結果のように、文字列を入力すると、文字列が英小文字および数字のみで構成されているかを判断するプログラムを書きなさい。ただし以下の条件を満たすこと。

- (1) ヘッダファイル ctype.h をインクルードし、文字列が英小文字および数字であるかの判断には全て ctype.h の標準ライブラリ関数を用いること
- (2) 文字列が英小文字および数字のみで構成されているかを判断する check 関数を定義し、main 関数で呼び出しをすること
- (3) check 関数の引数は void とし、戻り値として判断結果を main 関数に渡すこと。その時、最終的な判断結果の表示は main 関数で行うこと
- (4) 入力された文字列が英小文字および数字のみであった場合は入力が適切であることを表示し、それ以外が含まれていた場合は入力が不適切であることを表示すること

```
parallels@TA-Machine:~$ gcc -o 03 03.c
parallels@TA-Machine:~$ ./03
英小文字および数字を使った文字列を入力してください：abc333
入力は適切です
parallels@TA-Machine:~$ ./03
英小文字および数字を使った文字列を入力してください：Abc333
入力が不適切です。大文字は使用できません
parallels@TA-Machine:~$ ./03
英小文字および数字を使った文字列を入力してください：.abc333
入力が不適切です。使用できる文字は英数字のみです
parallels@TA-Machine:~$
```

問 04

次の実行結果のように、問 03 のプログラムで文字列が適切と判断された後に再度文字列を入力させ、文字列同士が一致するかどうかを判断するプログラムを書きなさい。ただし以下の条件を満たすこと。

- (1) ヘッダファイル string.h をインクルードし、文字列同士が一致するかの判断には全て ctype.h の標準ライブラリ関数を用いること
- (2) 文字列同士が一致するかを判断する関数を問 03 とは別に定義し、main 関数で呼び出しをすること
- (3) 後に入力された文字列が大文字であった場合、小文字に変換してから判断すること
- (4) 文字列同士が一致しなかった場合は入力が不適切であることを表示し、再度入力を促すようにすること

```
parallels@TA-Machine:~$ gcc -o 04 04.c
parallels@TA-Machine:~$ ./04
英小文字および数字を使った文字列を入力してください：abc333
入力は適切です
再度同じ文字列を入力してください
abc333
文字列が一致しました
parallels@TA-Machine:~$ ./04
英小文字および数字を使った文字列を入力してください：abc333
入力は適切です
再度同じ文字列を入力してください
abc331
文字列が一致しません再度同じ文字列を入力してください
abc333
文字列が一致しました
parallels@TA-Machine:~$
```

本日の課題はここまでです。1 ページ目の注意事項をよく読み、課題を提出してください。