

デバックについて

もし、コンパイル・リンクの時にエラー (Error) や警告 (Warning) メッセージが表示された場合には、プログラムの修正を行う必要があります。ただしエラーと警告メッセージでは、大きな違いがあります。

[エラー] コンパイルができない。

[警告] コンパイル・リンクはできる。実行も可能。しかし、表記が不適切。

という違いです。このため、警告メッセージであれば、意図したとおりにプログラムが動くこともあります。しかし、多くの場合は警告がないプログラムに書き換えることはできますので、演習では「要修正」として扱います。

それでは、エラーや警告メッセージを参考に、プログラムの修正をしてみましょう。とはいってもエラーメッセージが何を示しているか全然分からないと思います。エラーメッセージの意味を読み取るには多少の経験が必要です。

まずはエラーメッセージがどのように出るのかを説明します。エラーメッセージには、通常次の情報が出力されます。

- ・ エラー (警告) が見つかったプログラムのファイル名
- ・ エラー (警告) が見つかったプログラムの行
- ・ エラー (警告) の内容

bcpad でエラーが見つかった場合は以下のように表示されます。

The image shows a code editor window titled 'test.c' with the following code:

```
1 #include <stdio.h>↓
2 ↓
3 int main(void)↓
4 {↓
5     printf("Hello, world!");↓
6     ↓
7     return 0;□|
8 } [EOF]
```

Below the code editor is a 'メッセージ' (Message) window showing the compilation error:

```
C:\Users\take\Desktop> bcpad test.c
Borland C++ 5.0.2.0 (C:\Program Files\Borland\Borland C++\bin)
test.c:
エラー E2206 test.c 7: 不正な文字 ' ' (0x8140)(関数 main )
*** 1 errors in Compile ***
```

Annotations in the image explain the error message:

- 「エラーの内容 この場合は不正な文字があることを知らせている。」 points to the error text.
- 「ファイル名」 points to 'test.c' in the message.
- 「7行目でエラーが見つかったことを表す」 points to the line number '7' in the message.

ここで、「エラーメッセージの指す行数は、その場所に必ず誤りがあるのではなく、その周辺の場合がある」ということに注意してください。エラーが出た場合には、まず、その行に着目し、それでも見あたらなければ、次にエラーメッセージが示した行以前にエラーがあることが多いです。例えば { に対しての } を書き忘れたため、括弧の対応がくずれ、大きく離れたところにある別の } との対応がコンパイラに認識されたためにソースプログラムの最後の行をエラーとして指す場合もあります。

また、エラーが出た場合には、往々にして、多数のエラーメッセージが出てしまいます。これははじめの一つのエラーが原因で多くの箇所がエラーとみなされてしまうために起こってしまった現象です。そのため、一番はじめの行数のエラーメッセージからデバックを行います。

次の表は、現段階で誤りやすい箇所についてのエラーと警告メッセージとそれぞれの意味、修正方法の例です。

bcpad でのエラーメッセージ	エラーメッセージの意味	修正方法の例
不正な文字 ' '(0x8140)	全角文字がプログラムの不必要な部分に混ざっています。	削除するか、半角文字に修正します。全角のスペースは□で表示されているので、参考にしてください。
関数呼び出しに) がない または 複合文に } がない	括弧の対応がくずれています。	(と), { と } は対応する必要があります。数をそろえてください。
プロトタイプ宣言のない関数 'printq' の呼び出し	入力間違いです。	正しいものに修正します。この場合では printf です。

bcpad での警告メッセージ	警告メッセージの意味	修正方法の例
プロトタイプ宣言のない関数 'printf' の呼び出し	#include <stdio.h>が無いです。	#include <stdio.h>をプログラムの最初に書きます。