入門講習会 寄り道

デバッグとは、プログラムのバグを見つけ、それを修正することである。

ABC081のB問題を題材に、コンパイルエラーの見方や、簡単なデバッグの方法を紹介する。

1. コンパイルエラー

ABC081 Bを次のように書いたとする。

|  |
| --- |
| #include <stdio.h>    int main() {      int N;      int i;      int min;        scanf("%d", &N)      for(i = 0: i < N; i++) {          int A;          int cnt;          scanf("%d", &A);          while(A % 2 = 0) {              cnt++;              A /= 2;          }          if(min > cnt) min = cnt;      }      printf("%d\n", min);        return 0;  } |

これをtest.cという名前で保存し、gccでコンパイルすると、以下のようなエラーがでる。

|  |
| --- |
| test.c: In function 'main':  test.c:9:2: error: expected ';' before 'for'  for(i = 0: i < N; i++) {  ^~~ |

9:2というのは、9行目の2文字目がエラー箇所であることを表している。次の行で、その9行目2文字目の場所を表示してくれている。

エラーを意訳すると、「forの前に;がない」と書かれている。そこでforの手前、つまりscanfの()の後を見てみると、セミコロンがない。セミコロンを付け足せばこのエラーは消える。

修正してコンパイルすると、今度は別のエラーが出る。

|  |
| --- |
| test.c: In function 'main':  test.c:9:11: error: expected ';' before ':' token  for(i = 0: i < N; i++) {  ^  test.c:9:23: error: expected ';' before ')' token  for(i = 0: i < N; i++) {  ^  test.c:13:15: error: lvalue required as left operand of assignment  while(A % 2 = 0) {  　　　^ |

エラー内容を読んでもわからない場合は、該当箇所を睨んでみよう。それだけで修正できることが多い。今回の場合は、for(i = 0の直後に来ているのがセミコロンではなくてコロンになっている。9行目のエラーが2つ出ているが、セミコロンをコロンに直すだけで2つとも解決する。13行目のエラーはよく見ると、関係演算子==ではなく、代入演算子=になってしまっている。ここを直すとエラーは消える。

13行目についてもう少し詳しく説明しよう。代入演算子は変数に値を代入する演算子なのだから、(変数)=値の形で書かなければいけない。しかしA%2は定数のため、「(定数)=値ということはできないよ」とエラーを出していることになる。

==と=を書き間違えるミスはたまにある。しかも例えば次のように書いたとしたら、文法的には誤りではないためエラーは出ない。注意が必要だ。

 while(A = 0) { … }

1. 簡単なデバッグ方法(printfデバッグ)

さて、一通り修正が終わったコードは次のようになる。

|  |
| --- |
| #include <stdio.h>    int main() {      int N;      int i;      int min;        scanf("%d", &N);      for(i = 0; i < N; i++) {          int A;          int cnt;          scanf("%d", &A);          while(A % 2 == 0) {              cnt++;              A /= 2;          }          if(min > cnt) min = cnt;      }      printf("%d\n", min);        return 0;  } |

これを動かしてみると、サンプルの入力例1の時点ですでに間違えている。

|  |
| --- |
| 3 (入力)  8 12 40 (入力)  3 |

本来は2が出るはずなのに、3が出てしまっている。