

```
// 「Cで学ぶデータ構造とアルゴリズム」(西原清一) オーム社, 2008
// 図3・15(p.70)+図3・10(p.66)+解図3・B(p.211)
// BM法のメインプログラムとheur1(), heur2()
```

```
#include<stdio.h>
```

```
#define MAX 1000
```

```
int d[MAX], dd[MAX], f[MAX];
```

```
int n, m;
```

```
void heur1(char *pat)
```

```
{ int i;
```

```
  for (i=1; i <= MAX; i++) d[i]=m;
```

```
  for (i=1; i <= m; i++) d[pat[i]]=m-i;
```

```
// for (i=1; i<=256; i++) printf("d[%d]=%d, ", i, d[i]);
```

```
// printf("¥n");
```

```
}
```

```
void heur2(char *pat)
```

```
{ int j, k, t, w;
```

```
  for (k=1; k <= m; k++) dd[k]=2*m-k;
```

```
  j=m; t=m+1;
```

```
  while (j > 0) {
```

```
    f[j] = t;
```

```
    while (t <= m && pat[j] != pat[t]) {
```

```
      if (dd[t] > (w=m-j)) dd[t]=w;
```

```
      t=f[t];
```

```
    }
```

```
    j--; t--;
```

```
  }
```

```
  for (k=1; k <= m; k++) {
```

```
    if (dd[k] > (w=m+t-k)) dd[k]=w;
```

```
    if (k >= t) t=f[t];
```

```
  }
```

```
// for (w=1; w<=m; w++) printf("heur2: dd[%d]=%d¥n", w, dd[w]);
```

```
// printf("¥n¥n");
```

```
}
```

```

int bm(char *text, char *pat)
{
    int j,k;
    k=m;
    while (k <= n) {
        j=m;
        while (j > 0 && pat[j] == text[k]) {j--; k--;}
        if (j == 0) return k+1;          /* 照合成功 */
        else {
            if (d[text[k]] > dd[j]) k=k+d[text[k]];
            else k=k+dd[j];
        }
    }
    return -1;                          /* 照合失敗 */
}

main()
{
    int i;
    char *text = " ABABABBABABAABA", *pat = " BABAABA";
    n = 15; m = 7;

    printf("¥ntext=");
    for (i=0; i<=n+1; i++) printf(" %c", text[i]);
    printf("¥npat= ");
    for (i=0; i<=m+1; i++) printf(" %c", pat[i]);
    printf("¥n¥n");

    heur1(pat);
    heur2(pat);
    i = bm(text, pat);
    if (i != -1) printf("¥nFound at %d¥n", i);
    else printf("¥nNot found.¥n");
}

```