

# P1 werkcollege 3

algoritmen en programma's  
leer Karel rekenen  
C-strings

1

## schrijf getal

- als laatste bitje een 1 is: leg bal
- doe een stap
- deel getal door 2
  
- doe dit zolang het getal niet 0 is

2

## schrijfGetal in C++

```
void schrijfGetal ( int getal )  
{  
  while ( getal != 0 )  
  {  
    if ( getal % 2 == 1 )  
      legBal ( );  
    stap ( );  
    getal = getal / 2;  
  }  
}
```

merk op:  
we gebruiken een  
argument als variabele

3

## wat kan er mis gaan ?

- getal past niet in de wereld:
  - karel loopt in de muur
- oplossing: stop bij de muur
  - i.p.v. **while** ( getal != 0 )  
**while** ( ! muurVoor ( ) )
  - nadelen:
    - Karel loopt verder dan nodig bij klein getal
    - laatste positie wordt niet gebruikt !!

```
void schrijfGetal ( int getal )  
{ while ( getal != 0 )  
  { if ( getal % 2 == 1 )  
    legBal ( );  
    stap ( );  
    getal = getal / 2;  
  }  
}
```

4

## veilige schrijfGetal in C++

```
void schrijfGetal ( int getal )
{
  bool kanVerder = true;
  while ( getal != 0 && kanVerder )
  {
    if ( getal % 2 == 1 )
      legBal ( );
    if (! muurVoor ( ) )
      stap ( );
    else
      kanVerder = false;
    getal = getal / 2;
  }
}
```

5

## waarom moet dit moeilijk ?

- probleem: twee stappen
  - legBalAlsNodig
  - doeStap
- moet in deze volgorde
- stopconditie tussen stap 1 en 2
- lelijk:  
legBalAlsNodig();  
**while** ( ! voorMuur ( ) )  
{ doeStap ( ); legBalAlsNodig ( ); }

6

## speciaal statement in C++: break

```
void schrijfGetal ( int getal )
{
  while ( getal != 0 )
  {
    if ( getal % 2 == 1 )
      legBal ( );
    if ( muurVoor ( ) )
      break;
    stap ( );
    getal = getal / 2;
  }
}
```

verlaat huidige loop  
onmiddellijk  
Ga na de loop verder

7

## meer van hetzelfde

- een soort gelijk probleem hebben we natuurlijk ook bij optellen, aftrekken etc.
- we zullen het niet zwaar aanrekenen als de laatste bal niet goed is!

8

## optellen

- twee mogelijkheden
  - gebruik logische expressies om te bapalen of er een bal moet komen  
if ( XOR( carry, XOR ( digit1, digit2 )))  
legBalZeker ( );
  - gebruik een tellertje voor aantal ballen  
if ( aantal % 2 == 1 )  
legBal ( );
- keuze is een kwestie van smaak

9

## optellen in C++

```
void telop ( )  
{ naarRechtsBoven ( );  
  int aantal = 0;  
  while ( true )  
  { linksom ( );  
    for ( int r=0; r<2; r=r+1 )  
    { if ( opBal ( ) )  
      aantal = aantal+1;  
      stap ( );  
    }  
    if ( aantal % 2 == 1 )  
    legBal ( );  
    aantal = aantal / 2;  
    rechtsom ( );  
    if ( muurVoor ( ) )  
      break;  
    stap ( );  
    naarBoven ( 2 );  
  }  
}
```

tot St. Jutemus

bekijk alle rijen

dit bit

bepaal carry

verlaat eindeloze loop

10

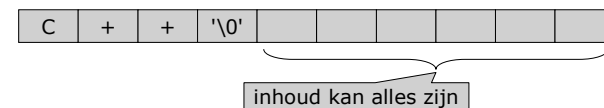
## opgave 3

C-strings  
vervanging van patroon

11

## strings in C++

- C++ kent verschillende soorten strings
- we gebruiken nu C-strings:  
een rij van letters
  - per afspraak volgt er een '\0' na het laatste teken
  - rij kan dus 'te lang' zijn
  - char voorbeeld [10] = "C++"



12

## opdracht 3

- maak 'n functie  
vervangAlle(char string [], char oud [], char nieuw [])
- string wordt veranderd
- bijvoorbeeld  
char voorbeeld [10] = "C++";  
vervangAlle(voorbeeld,"+","--");  
cout << voorbeeld << endl;
- geeft "C----"

13

## aanpak

- splits in kleinere onderdelen
- **while** ( nietAanEinde )  
vervang ( string, oud, nieuw, pos )
- voeg argumenten tot voor communicatie tussen functies

14

## aanpak 2

- vervang ( string, oud, nieuw, pos )
- ook splitsen in onderdelen
  - zoek patroon
  - pas ruimte aan
  - kopieer nieuw naar de juiste positie
- voeg weer argumenten en functieresultaten toe voor communicatie tussen functies
  - b.v. positie waarop patroon gevonden is
  - -1 betekent niet gevonden

15

## voorwaarden & randgevallen

- mag string leeg zijn?
  - en oud en nieuw?
- nieuw langer dan oud
  - nieuw korter dan oud
- patroon aan begin
- patroon aan eind
- patroon komt niet voor
- patroon komt vaak voor

16

## testen

- schrijf een programma dat alle speciale gevallen test
  - print wat je gaat doen
  - print het resultaat van testen
- vast handig om rijtje van strings te gebruiken

```
const int StrLengte = 80;  
char strings [2][StrLengte]  
= { "Algoritmen en programma's."  
  , ""  
  };
```

17

## het nieuwe project

- maak nu geen windows application, maar een **console application**
- gebruik de main file die je krijgt!
  - bevat bibliotheek die je nodig hebt
  - main functie die 'niets' doet

```
int main( int argc, char *argv[] )  
{  
    jouw werk hier  
    argumenten van programma  
    system("PAUSE");  
    wacht op any key  
    return EXIT_SUCCESS;  
    vertel windows dat alles goed was  
}
```