TFY4235/FYS8904

Solution problemset 2 Spring 2015

NTNU

Institutt for fysikk

Solutions to the problems are provided in the *code* subdirectory in both Fortran and C.

Problem 1.

The following program uses the Fortran 90 option to simply write the bit contents of the real number a to the screen:

Listing 1: rform.f

```fortran
      program rform
c Fortran 90 output format
      write(*,*) 'input='
      read (*,*) a
      write(*,1) a
1     format(1x,b32)
      end
```

Rather than using the specific Fortran 90 bit format, (”b32” in the format statement), we may use bit functions to assemble the numbers. Here are three Fortran programs that read the bit representation of the machine you are using for integers, real and real*2 numbers using this technique. Note in particular what happens with the numbers that are transfered to the subroutine in programs two and three. I am here using a ‘dirty’ trick that sometimes can come in very handy: I am reading real numbers into the subroutine, while inside the subroutine the numbers are treated as integers. The same trick can come in very handy when dealing with complex numbers: Complex numbers can be treated as such on the ouside of a subroutine and as vectors of length two on the inside — or vice versa.

Such dirty tricks are very useful, but you need someone to tell you about them. I just did.

Program reading the machine’s integer representation.

Listing 2: heltall.f

```fortran
      program heltall
c Integer representation
      dimension in(32)
      write(*,*) 'input='
      read (*,*) ia
      do i=1,32
      in(i)=iand(1,ishft(ia,1-i))
      enddo
      write(*,1) (in(i),i=32,1,-1)
1     format(1x,32(i1))
      end
```

Program reading the machines real number representation.

Listing 3: realtall.f

```fortran
      program realtall
c Real representation
2     continue
      write(*,*) 'real ='
      read (*,*) a
      call rint(a,b)
      write(*,*) b
      goto 2
      end
c
      subroutine rint(ia,ib)
      dimension in(32)
      do i=1,32
      in(i)=iand(1,ishft(ia,1-i))
      enddo
      write(*,1) (in(i),i=32,1,-1)
1     format(1x,32(i1))
      ib=ia
      return
      end
```

Program reading the machine's real*2 number representation.

Listing 4: dobbelta.f

```fortran
      program dobbeltall
      double precision a,b
c Double representation
2     continue
      write(*,*) 'Double ='
      read (*,*) a
      call dint(a,b)
      write(*,*) b
      goto 2
      end
c
      subroutine dint(a,b)
      dimension a(2),b(2)
      call rint(a,b)
      return
      end
c
      subroutine rint(ia,ib)
      dimension in(2,32),ia(2),ib(2)
      do j=1,2
      do i=1,32
      in(j,i)=iand(1,ishft(ia(j),1-i))
      enddo
      enddo
      write(*,1) ((in(j,i),i=32,1,-1),j=1,2)
1     format(1x,64(i1))
      ib(1)=ia(1)
      ib(2)=ia(2)
      return
      end
```

Problem 2.

Here is a program measuring the precision of your machine.

Listing 5: presisjo.f

```
      program presisjon
c Maaler maskinens presisjon
      pres=1.
      do i=1,1000
      pres=pres*0.5
      prp1=1.+pres
      if(prp1.le.1.) goto 100
      enddo
100   pres=pres*2.
      write(*,*) pres
      end
```

And, now you can use these programs to find out how your computer stores numbers.