

```

{*****
{                                C O N F I G P                                }
{*****}
{* Task : Displays the configuration of the PC. *}
{*****}
{* Author : Michael Tischer *}
{* Developed on : 07/07/87 *}
{* Last update : 01/29/92 *}
{*****}

```

```

program CONFIGP;

```

```

Uses Crt, Dos; { Add CRT and DOS units }

```

```

{*****}
{* PrintConfig: Displays PC configuration. *}
{* Input : None *}
{* Output : None *}
{* Info : Configuration varies with the type of PC. *}
{*****}

```

```

procedure PrintConfig;

```

```

var AT : boolean; { Is PC an AT? }
    Regs : Registers; { Processor registers for interrupt call }

```

```

begin
  ClrScr; { Clear screen }
  if Mem[$F000:$FFFF] = $FC then AT := true { Test for AT, }
  else AT := false; { PC or XT }
  writeln('CONFIGP - (c) 1987, 1992 by Michael Tischer');
  writeln;
  writeln('Your PC Configuration ');
  writeln('-----');
  write('PC Type : ');
  case Mem[$F000:$FFFF] of { Read PC type and display }
    $FF : writeln('PC'); { $FF (FFH) is a PC }
    $FE : writeln('XT'); { $FE (FEH) is an XT }
    else writeln('AT or higher') { $FC (FCH) is an AT }
  end;

```

```

  Intr($12, Regs); { RAM from BIOS interrupt }
  writeln('Conventional RAM : ', Regs.ax, ' K');
  if AT then { Is the PC an AT? }
  begin { Yes }
    Regs.ah := $88; { Read function number for extended memory }
    Intr($15, Regs); { Call BIOS cassette interrupt }
    writeln('Additional RAM : ', Regs.ax, ' K over 1 megabyte');
  end;
  Intr($11, Regs); { Call BIOS configuration interrupt }
  write('Default video mode : ');
  case Regs.al and 48 of { Get video mode }
    0 : writeln('Undefined');
    16 : writeln('40x25 character color card');
    32 : writeln('80x25 character color card');
    48 : writeln('80x25 character mono card');
  end;
  writeln('Disk drives : ', succ(Regs.al shr 6 and 3));
  writeln('Serial interfaces : ', Regs.ah shr 1 and 3);
  writeln('Parallel interfaces : ', Regs.ah shr 6);
end;

```

```

{*****}
{* MAIN PROGRAM *}
{*****}

```

```

begin
  PrintConfig; { Display configuration }
end.

```