

David Pereira Da Silva - 58893 - 07-05-2010

Folha de Calculo

	A	B	C	D	E	...
1						
2						
3						
⋮						

formula
 texto
 numeros

typedef struct cell

```

{ int tipo;
  union uContent
  { Exp expression;
    char * text;
    int number;
    double dnumber;
  } content;
} * cell;

```

int linha, coluna;

typedef struct sline

```

{ cell c;
  struct sline * next;
} * line, Nline;

```

typedef struct sSS

SpreadSheet

```

{ Line l;
  struct sSS * next;
} * SS, NSS;

```

void showcell (cell c)

{ switch (c.tipo)

```

{ case EXP = showExp (c.content.expression);
  break;

```

```

case TEXT = printf ("%s", c.content.text);
  break;

```

```

case INTEGER = printf ("%d", c.content.number);
  break;

```

```

case REAL = printf ("%2f", c.content.dnumber);
  break;

```

```

}
}

```

```

void showline (line l)
{
  if (l)
    { showline (l->c);
      showline (l->next);
    }
}

```

```

cell conscellexp (exp e)
{
  cell aux;
  aux->tipo = EXP;
  aux->content->expression = e;
  return aux;
}

```

```

SS consSS (SS o, line l)
{
  SS aux;
  if (!s || (l->c.linha < s->l->c.linha))
    { aux = (SS) malloc (sizeof(SS));
      aux->l = l;
      aux->next = s;
      return aux;
    }
  else
    { s->next = consSS (s->next, l)
      return s;
    }
}

```

```

void showSS (SS s)
{
  if (s)
    { showline (s->l);
      showSS (s->next);
    }
}

```

```

line consline (line l, cell c)
{
  line aux;
  if (!l || (c->coluna < l->c->coluna))
    { aux = (line) malloc (sizeof(line));
      aux->c = c;
      aux->next = l;
      return aux;
    }
  else
    { l->next = consline (l->next, c);
      return l;
    }
}

```