# CSCV 453: B-- Language Specification

## Extended BNF Notation

- Alternatives are separated by vertical bars: i.e., `*a* | *b*' stands for ``*a* ***or*** *b*''.
- Square brackets indicate optionality: `[ *a* ]' stands for an optional *a*, i.e., ``*a* | *epsilon*'' (here, *epsilon* refers to the empty sequence).
- Curly braces indicate repetition: `{ *a* }' stands for ``*epsilon* | *a* | *aa* | *aaa* | ...''

## 1. Lexical Rules

| | | |
|---|---|---|
| *letter* | ::= | a \| b \| ... \| z \| A \| B \| ... \| Z |
| *digit* | ::= | 0 \| 1 \| ... \| 9 |
| **id** | ::= | *letter* { *letter* \| *digit* \| _ } |
| **intcon** | ::= | *digit* { *digit* } |
| **charcon** | ::= | **'***ch***'** \| **'\n'** \| **'\0'**, where *ch* denotes any printable ASCII character (as specified by **isprint()**) other than **** (backslash) and **'** (single quote). |
| **stringcon** | ::= | **"**{ *ch* }**"**, where *ch* denotes any printable ASCII character (as specified by **isprint()**) other than **"** (double quotes) and newline. |
| *Comments* | | Comments are as in C, i.e. a sequence of characters preceded by ***/**** and followed by ****/***, and not containing any occurrence of ****/***. |

## 2. Syntax Rules

Nonterminals are shown in italics; terminals are shown in boldface, and sometimes enclosed within quotes for clarity.

### 2.1 Grammar Productions

*prog* : { *stmt* }

*stmt* : **if** `**(**' *expr* `**)**' *stmt* [ **else** *stmt* ]

| **while** `**(**' *expr* `**)**' *stmt*

| **for** `**(**' [ *assg* ] `**;**' [ *expr* ] `**;**' [ *assg* ] `**)**' *stmt*

| **return** [ *expr* ] `**;**'

| *assg* `**;**'

| **id** `**(**' [*expr* { `**,**' *expr* } ] `**)**' `**;**'

| `**{**' { *stmt* } `**}**'

| `**;**'

*assg* : **id** [ `**[**' *expr* `**]**' ]`**=**' *expr*

*expr* : '**-**' *expr*

| '**!**' *expr*

| *expr binop expr*

| *expr relop expr*

| *expr logical_op expr*

| **id** [ `**(**' [*expr* { `**,**' *expr* } ] `**)**' | `**[**' *expr* `**]**' ]

| `**(**' *expr* `**)**'

| **intcon**

| **charcon**

| **stringcon**

*binop* : **+**

| **-**

| *****

| */*

*relop* : **==**

| | **!=**

| | **<=**

| | **<**

| | **>=**

| | **>**

*logical_op* : **&&**

| **||**

## 2.2. Operator Associativities and Precedences

The following table gives the associativities of various operators and their relative precedences. An operator with a higher precedence binds ``tighter'' than one with lower precedence. Precedences decrease as we go down the table.

| **Operator** | **Associativity** |
|---|---|
| **!, -** (unary) | right to left |
| ***, /*** | left to right |
| **+, -** (binary) | left to right |
| **<, <=, >, >=** | left to right |
| **==, !=** | left to right |
| **&&** | left to right |
| **\|\|** | left to right |
| | |