Time: 4 to 5 mins.

* Slide 1: (15-30 seconds)
* The goal of ACP is to create a minimalistic language with basic arithmetic, Boolean and stack functionalities along with function implementations. Creating the language will help the team understand the design principles of programming language and also the implementation issues in designing one.
* We are team number 6 and we will be explaining the //flow of the presentation
* Slide 2: (30 seconds)

The ACP language has 3 supported data types. The integer type is used with the keyword ‘int’ followed by an ID. A value HAS to be assigned to the ID at the time of declaration.

Boolean values are assigned with ‘YES’ or ‘NO’ values.

Stack in ACP has 3 functions: push, pop and show. Once the stack is declared, a number value or a Boolean value can be pushed into the stack

Slide 3: ( 20 seconds)

The arithmetic operators in ACP are similar to the C language. The 4 arithmetic operators in the language are multiplication, addition, subtraction and division. As you can see from the giving examples, these operators can be used with terms or variables or a combination of both.

Slide 4: (30 seconds)

* There are 5 relational operators in ACP. Each of these five relational operators take in two operands. These two operands must both be *arithmetic*.
* The outcome of a comparison is a LOGICAL value. In the first example the values of x1 and x2 are compared and the result is ‘NO’ because the values aren’t equal.
* In the second example the outcome is evaluated only after the arithmetic operations are evaluated. If not, there are no values to compare.

Slide 5: (30 seconds)

This slide shows the operator precedence in the language. In Arithmetic operations Multiplication and division have higher precedence over addition and subtraction have the same precedence. Operands with the same precedence are evaluated left to right.

The 5 relational operators all have the same order of preference.

The example given shows that the outcome of the expression is possible only after the arithmetic operators are evaluated. The arithmetic operators have higher order of precedence over the relational operators.

Slide 6: (35 seconds)

* Functions in ACP can be used with function parameters or without them and also it can be just a block of code. It always begins with the keyword ‘function’ .
* A regular block of code in ACP can be an If statement ,While statement
* Call ,Return statement, Print statement, Stack declaration ,Stack operations.
* Recursion can be used with functions and as shown in the code snippet, we can call a function from within it. Every recursive method must have a base case, a condition under which no recursive call is made, to prevent infinite recursion.
* Slide 7: (15 seconds)
* ACP supports the use of if and else statements. Also, the if and else statements can be nested.
* The if statement in ACP can be used with or without the else statement

All relational operators can be used in the condition part of the if function.

Slide 8: (25 seconds)

While function is the only looping function available in ACP. It is similarly structured to that of the C language. The syantax is the keyword while following the condition and the block of statements. Sample code shows the implementation of while.

ACP is a block-structured language where a ‘<<’ introduces a new scope. Scopes are nested when a new variable is declared in an inner scope.