

# Set Cover Problem

## Steps to run the project

1. Open index.html with any browser (Chrome recommended)
2. Enter the Universal set and subsets in the area provided on the screen and press enter.
3. You can enter the subsets as many as you want as it is made dynamic.
4. After the input is given, enter space to see the animation of steps of the algorithm.
5. Stop when the universal set is filled with green colour completely.
6. You got the required minimum number of subsets on the right side of the screen.

## Source Code

Index.html

```
<!DOCTYPE html>
<html>
<head>
<title>SetCover Problem</title>
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.0/css/bootstrap.min.css">
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.0/js/bootstrap.min.js"></script>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/animatejs/2.2.0/animate.js"></script>
<link rel="stylesheet" href="style.css">
<script src="script.js"></script>
</head>
<body>
  <h1 class="main-heading">SET COVER PROBLEM</h1>

  <div class="row">
    <div class="col">
```

```

<div id="uniset" class="form-group">
  <input class="form-control set-inputs"
    v-model="newUniset"
    v-on:keyup.enter="addUniset"
    placeholder="Enter Universal Set"
  >
  <p id="uniset-clc" class="uniset-clc" v-for="set in uniset">{{ set.data }}</p>
</div>
</div>

<div class="col">
<div id="subsets" class="form-group">
  <input class="form-control set-inputs"
    v-model="newSubset"
    v-on:keyup.enter="addNewSubset"
    placeholder="Enter Subsets"
  >
  <p class="subsets-clc" v-bind:id="index" v-for="(subset,index) in subsets">{{ subset.data
}} </p>
</div>
</div>
<div class="col">
  <h2 class="subsets-heading mid-head">INTERMEDIATE SELECTION</h2>
</div>
<div class="col">
  <h2 class="subsets-heading">FINAL SUBSETS</h2>
</div>

</div>

<script>

```

```

var subsets_list = [];
var universalSet = [];
var answer = [];
var anpos = [];
var universalSet_3 = [];
var midtran = [];
var completedElements = [];
var n;

```

```

var vueinst = new Vue({
  el: '#subsets',
  data: {

```

```

    newSubset: "",
    subsets: []
  },
  methods: {
    addNewSubset: function () {
      this.subsets.push({
        data: this.newSubset
      })
      subsets_list.push(this.newSubset.split(',').map(Number))
      this.newSubset = ""
    }
  }
})

```

```

var unisetVue = new Vue({
  el: '#uniset',
  data:{
    newUniset: "",
    uniset: []
  },
  methods:{
    addUniset: function(){
      this.uniset.push(
        {
          data: this.newUniset
        })
      var arr = this.newUniset.split(',').map(Number)
      for(var i=0;i<arr.length;i++){
        universalSet.push(arr[i])
      }

      this.newUniset = ""
    }
  }
})

```

```

var common = function(a,b){

  var result = [];

```

```

for(var i=0;i<a.length;i++){

    for(var j=0;j<b.length;j++){

        if(a[i] === b[j]){
            result.push(a[i]);
            break;
        }
    }
}
return result;
}

```

```

var algo = function ()
{
    var subsets_list_2;
    var count;
    var position_2;
    var universalSet_2;
    var position;
    var subcount;

    count = 0;
    subsets_list_2 = subsets_list.slice();
    universalSet_2 = universalSet.slice();
    subcount = subsets_list.length;
    n = universalSet.length;

    while(count < n){

        var clen = 0;
        var pos = 0;
        console.log("Subset count: " + subsets_list.length);
        var midd = [];
        for(var i=0;i<subsets_list.length;i++){

            var common_elements = common(universalSet,subsets_list[i]);
            var tlen = (common_elements.length);
            console.log(tlen);

            if(clen < tlen){
                clen = tlen;
            }
        }
    }
}

```

```

        pos = i;
        midd.push(subsets_list_2.indexOf(subsets_list[pos]));
    }

    }
    midtran.push(midd);
    position = pos;
    console.log("postion "+position);
    console.log("Subset value" + subsets_list[pos]);
    if(pos > -1){
        position_2 = subsets_list_2.indexOf(subsets_list[pos]);
    }
    console.log("postion2 "+position_2);
    temp = common(universalSet,subsets_list[pos]);
    console.log(temp)
    answer.push(subsets_list[pos]);
    subsets_list = []
    for(var i=0;i<subsets_list_2.length;i++){

        if(i!=pos){

            subsets_list.push(subsets_list_2[i]);
        }

    }

    for(var i=0;i<clen;i++){
        universalSet=[]
        for(var j=0;j<universalSet_2.length;j++)
        {
            var flag=0;
            for(var k=0;k<temp.length;k++)
            {
                if(universalSet_2[j]==temp[k])
                {
                    flag=1;
                }
            }
            if(flag==0)
            {
                universalSet.push(universalSet_2[j]);
            }
        }

        count++;
    }

```

```

    }

    universalSet_2 = universalSet.slice();
    universalSet_3.push(universalSet_2);

    console.log(universalSet);
    anpos.push(position_2);
    completedElements.push(count);
    console.log("Count" + count);

  }
}

```

```

var count1=0;
var count2=0;
var count3=0;
document.querySelector('html').onkeyup = function(e){

    e.preventDefault();

    if(e.keyCode == 32){
        if(count1==0)
        {
            algo();
            count1++;
        }
        if(count2<anpos.length)
        {
            console.log("midtran: "+midtran[count2]);
            if(count3<midtran[count2].length)
            {
                if(count3!=0)
                {
                    var box = anime({
                        targets: [document.getElementById(midtran[count2][count3-1].toString())],
                        translateX:{
                            value: 0,
                            duration: 1000
                        }
                    })
                }
                console.log("back"+midtran[count2][count3-1]);
            }
        }
    }
}

```

```

    var box = anime({
    targets: [document.getElementById(midtran[count2][count3].toString())],
    translateX:{
        value: 500,
        duration: 1000
    }
    })
    console.log("mid"+midtran[count2][count3]);
    count3++;
}

if(count3==midtran[count2].length+1){
    var x;
    var y = anpos[count2];
    x=(completedElements[count2]/n)*100;
    var box = anime({
    targets: [document.getElementById(anpos[count2].toString())],
    translateX:{
        value: 850,
        duration: 1000
    }
    }).finished.then(=>{

        $('#'+y).css('background','linear-gradient(to right, #ade89b 0%,#ade89b
'+x+'%,#ade89b '+x+'%,#ade89b 100%)');

        $('#uniset-cls').css('background','linear-gradient(to right, #ade89b 0%,#ade89b
'+x+'%,#f26060 '+x+'%,#f26060 100%)');
    })
    console.log("front"+midtran[count2][count3-1]);
    count2++;
    count3=0;
}
if(count3 == midtran[count2].length)
{
    count3++;
}

}
}
}

</script>
</body>
</html>

```

## Style.css

```
body{

    background-image: url('background.jpg');

}

.main-heading{
    text-align: center;
    margin-top: 20px;
    color: #f2fbff;
}

.set-inputs{

    margin-top: 20px;
    width: 300px;
    color: #b5dcf2;
}

#subsets{

    margin-left: 20px;
}

.uniset-cls{

    border-radius: 50px;
    background-position: left top;
    background-repeat: repeat;
    padding-left: 20px;
    padding-bottom: 60px;
    padding-top: 60px;
    padding-right: 20px;
    width: 170px;
    height: 170px;
    background-color: #f26060;
    margin-top: 30px;
    text-align: center;
```



```
vertical-align: middle;

}

.subsets-cls{

border-radius: 50px;
background-position: left top;
background-repeat: repeat;
padding: 20px;
width: 120px;
height: 120px;
padding-left: 20px;
padding-bottom: 40px;
padding-top: 40px;
padding-right: 20px;
background-color:#b5dcf2;
margin-top: 30px;
text-align: center;
vertical-align: middle;

}

#uniset{

margin-left: 20px;

}

#uniset-cls{
margin-top: 30px;

}

.subsets-heading{
color: #f1f1f1;
margin-top: 20px;
}
```

```

.content {
  opacity: 0;
  display: -webkit-box;
  display: -ms-flexbox;
  display: flex;
  -webkit-box-orient: horizontal;
  -webkit-box-direction: normal;
  -ms-flex-direction: row;
  flex-direction: row;
  width: 100%;
  height: 100%;
  transition: opacity 1s ease;
}

body.ready .content {
  opacity: 1;
}

/* Headers */

.header {
  position: absolute;
  z-index: 1;
  top: 0;
  width: 239px;
  padding: 0;
}

.logo {
  display: -webkit-box;
  display: -ms-flexbox;
  display: flex;
  -webkit-box-align: center;
  -ms-flex-align: center;
  align-items: center;
  width: 240px;
  height: 60px;
  padding: 10px 0 10px 55px;
  color: white;
  text-decoration: none;
  background-image: url(../img/icons/icon-doc.png);
  background-size: 32px;
  background-position: 12px 13px;
}

```

```

    background-repeat: no-repeat;
}

header {
    position: -webkit-sticky;
    position: -moz-sticky;
    position: -o-sticky;
    position: -ms-sticky;
    position: sticky;
    top: 0;
    z-index: 2;
    display: -webkit-box;
    display: -ms-flexbox;
    display: flex;
    -webkit-box-align: center;
    -ms-flex-align: center;
    align-items: center;
    -webkit-box-pack: justify;
    -ms-flex-pack: justify;
    justify-content: space-between;
    width: 100%;
    height: 60px;
    padding-left: 20px;
    padding-right: 20px;
    border-bottom: 1px solid #000;
}

.logo, h2 {
    opacity: .6;
    font-size: 22px;
    letter-spacing: 2px;
    text-transform: uppercase;
    font-weight: 200;
}

.logo:hover {
    opacity: 1;
}

/* Panes */

.pane {
    position: relative;
    height: 100%;
}

```

```
/* Sidebar */
```

```
.sidebar {  
  width: 240px;  
  padding-top: 60px;  
  border-right: 1px solid #000;  
}
```

```
.navigation {  
  overflow-y: auto;  
  width: 100%;  
  height: 100%;  
  padding: 10px 0 20px 20px;  
}
```

```
.navigation ul {  
  height: 30px;  
  overflow-y: hidden;  
  -webkit-transition: opacity .4s ease;  
  transition: opacity .4s ease;  
}
```

```
.navigation ul li {  
  position: relative;  
}
```

```
.navigation ul li:before {  
  content: "";  
  display: block;  
  position: absolute;  
  top: 3px;  
  left: 2px;  
  width: 1px;  
  height: 14px;  
  background-color: white;  
}
```

```
.navigation ul li:first-child:before {  
  display: none;  
}
```

```
.navigation a {  
  display: block;  
  padding-left: 15px;
```

```
color: #FFF;  
text-decoration: none;  
font-size: 12px;  
line-height: 20px;  
text-transform: uppercase;  
}
```

```
.navigation ul li:first-child a {  
padding-left: 0;  
padding-top: 10px;  
padding-bottom: 10px;  
}
```

```
.navigation ul li a,  
.navigation ul li:before {  
opacity: .4;  
-webkit-transition: opacity .4s ease, color .4s ease;  
transition: opacity .4s ease, color .4s ease;  
}
```

```
.navigation ul.active a {  
opacity: .7;  
}
```

```
.navigation ul li:hover a,  
.navigation ul li:hover:before {  
-webkit-transition: none;  
transition: none;  
opacity: 1;  
}
```

```
.navigation ul li.active a,  
.navigation ul li.active:before {  
opacity: 1;  
color: currentColor;  
}
```

```
.navigation ul li.active:before {  
background-color: currentColor;  
}
```

```
.navigation ul.active li:first-child a,  
.navigation ul.active li:first-child a:hover {  
opacity: 1;  
color: currentColor;
```

```
}
```

```
/* Code */
```

```
.code-examples {  
  width: 50%;  
  -webkit-box-orient: vertical;  
  -webkit-box-direction: normal;  
  -ms-flex-direction: column;  
  flex-direction: column;  
  border-left: 1px solid #000;  
}
```

```
.code-pane {  
  position: relative;  
  border-bottom: 1px solid #000;  
}
```

```
.code-pane-js {  
  height: 66%;  
}
```

```
.code-pane-html {  
  height: 34%;  
}
```

```
.code-header {  
  position: absolute;  
  top: 0;  
}
```

```
pre {  
  overflow: auto;  
  font-family: 'Space Mono', monospace;  
  line-height: 22px;  
}
```

```
.code-output {  
  width: 100%;  
  height: 100%;  
  padding: 80px 20px 20px 20px;  
}
```

```
/* Demos */
```

```
.demos {  
  overflow-y: scroll;  
  width: calc(50% - 240px);  
  min-width: 330px;  
}
```

```
article {  
  width: 100%;  
  display: -webkit-box;  
  display: -ms-flexbox;  
  display: flex;  
  -webkit-box-orient: vertical;  
  -webkit-box-direction: normal;  
    -ms-flex-direction: column;  
      flex-direction: column;  
  -ms-flex-wrap: wrap;  
    flex-wrap: wrap;  
  -webkit-box-align: start;  
    -ms-flex-align: start;  
      align-items: flex-start;  
  -webkit-box-pack: start;  
    -ms-flex-pack: start;  
      justify-content: flex-start;  
}
```

```
.demos-title {  
  opacity: 1;  
}
```

```
.demo {  
  position: relative;  
  z-index: 1;  
  display: -webkit-box;  
  display: -ms-flexbox;  
  display: flex;  
  -webkit-box-align: center;  
    -ms-flex-align: center;  
      align-items: center;  
  -webkit-box-pack: center;  
    -ms-flex-pack: center;  
      justify-content: center;  
  width: 100%;  
  min-height: 200px;  
  padding-top: 35px;  
  border-bottom: 1px solid #000;
```

```
    cursor: pointer;
}

.demo.controls {
    padding-top: 50px;
}

.demo.active.controls {
    cursor: default;
}

.demo:before {
    content: "";
    position: absolute;
    z-index: 0;
    top: 0;
    left: 0;
    opacity: 0;
    display: block;
    width: 100%;
    height: 100%;
    background-color: currentColor;
}

.demo.active:before {
    opacity: .05;
}

.demo-content {
    position: relative;
    display: -webkit-box;
    display: -ms-flexbox;
    display: flex;
    -webkit-box-orient: vertical;
    -webkit-box-direction: normal;
    -ms-flex-direction: column;
    flex-direction: column;
    -webkit-box-pack: center;
    -ms-flex-pack: center;
    justify-content: center;
    text-align: center;
    width: 290px;
    height: 100%;
}
```



```
#penner .demo-content {  
  padding: 25px 0;  
}
```

```
#timeline .demo-content,  
#motionpath .demo-content {  
  padding: 40px 0;  
}
```

```
.demo-title {  
  position: absolute;  
  top: 0;  
  left: 20px;  
  width: calc(100% - 40px);  
  padding: 23px 0;  
  font-size: 12px;  
  letter-spacing: 2px;  
  text-transform: uppercase;  
  color: #FFF;  
  opacity: .7;  
}
```

```
.active .demo-title {  
  color: currentColor;  
  opacity: 1;  
}
```

```
.line {  
  width: 100%;  
  padding: 1px 0px;  
}
```

```
.player {  
  opacity: .4;  
  display: -webkit-box;  
  display: -ms-flexbox;  
  display: flex;  
  -webkit-box-align: center;  
  -ms-flex-align: center;  
  align-items: center;  
  -webkit-box-pack: center;  
  -ms-flex-pack: center;  
  justify-content: center;  
  padding-top: 20px;  
  padding-bottom: 10px;
```

```
}
```

```
.player button {  
  position: relative;  
  display: block;  
  margin: 0 0 0 -1px;  
  padding: 5px 10px;  
  font-size: 12px;  
  text-transform: uppercase;  
  border: 1px solid currentColor;  
  background-color: transparent;  
  color: currentColor;  
}
```

```
.player button:focus {  
  outline: none;  
}
```

```
.active .player {  
  opacity: 1;  
}
```

```
.active .player button:hover:after {  
  content: " ";  
  opacity: .3;  
  position: absolute;  
  left: 0;  
  top: 0;  
  display: block;  
  width: 100%;  
  height: 100%;  
  background-color: currentColor;  
}
```

```
.player button:first-child {  
  margin-left: 0;  
}
```

```
.player input {  
  margin-left: 10px;  
}
```

```
.text-output {  
  padding-top: 10px;  
  padding-bottom: 10px;
```

```
font-size: 16px;  
color: currentColor;  
text-align: center;  
background-color: transparent;  
border: 1px solid currentColor;  
}
```

```
.log {  
width: 100%;  
height: 22px;  
background: transparent;  
color: currentColor;  
border: none;  
font-size: 16px;  
text-align: center;  
}
```

```
.logs {  
pointer-events: none;  
display: -webkit-box;  
display: -ms-flexbox;  
display: flex;  
-webkit-box-orient: vertical;  
-webkit-box-direction: normal;  
-ms-flex-direction: column;  
flex-direction: column;  
-webkit-box-align: center;  
-ms-flex-align: center;  
align-items: center;  
position: absolute;  
width: 100%;  
height: 100%;  
}
```

```
.log:focus {  
outline: none;  
}
```

```
input.progress {  
cursor: ew-resize;  
}
```

```
.square,  
.circle {  
pointer-events: none;
```

```
position: relative;
width: 28px;
height: 28px;
margin: 1px;
background-color: currentColor;
font-size: 12px;
}
```

```
.circle {
border-radius: 50%;
}
```

```
.triangle {
pointer-events: none;
position: relative;
width: 0;
height: 0;
border-style: solid;
border-width: 0 14px 24px 14px;
border-color: transparent transparent currentColor transparent;
}
```

```
.small {
width: 18px;
height: 18px;
}
```

```
.large {
width: 48px;
height: 48px;
}
```

```
.stretched {
height: 2px;
}
```

```
.shadow {
position: absolute;
opacity: .2;
}
```

```
.motion-path {
position: relative;
width: 256px;
margin: auto;
}
```

```
}
```

```
.follow-path {  
  position: absolute;  
  top: -9px;  
  left: -9px;  
}
```

```
.align-center {  
  -webkit-box-align: center;  
  -ms-flex-align: center;  
  align-items: center;  
}
```

```
#cssProperties .el {  
  -webkit-transform: translate3d(0,0,0);  
  transform: translate3d(0,0,0);  
}
```

```
/* Responsive */
```

```
.header {  
  width: 100%;  
}
```

```
.sidebar,  
.code-examples {  
  display: none;  
}
```

```
.demos {  
  top: 60px;  
  width: 100%;  
  min-width: auto;  
  height: calc(100% - 60px);  
  -webkit-overflow-scrolling: touch;  
}
```

```
@media (min-width: 600px) {  
  .header {  
    width: 239px;  
  }  
  .sidebar {  
    display: inherit;  
  }  
}
```

```
.demos {  
  top: 0;  
  width: calc(100% - 240px);  
  height: 100%;  
}  
}
```

```
@media (min-width: 1100px) {  
  .code-examples {  
    display: inherit;  
  }  
  .demos {  
    width: calc(50% - 240px);  
  }  
}
```